1990-09

Automatic satellite image navigation

Spaulding, Brian C.

Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/34940

2

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AUTOMATIC SATELLITE IMAGE
NAVIGATION

by

Brian C. Spaulding

September 1990

Thesis Advisor Professor C.H. Wash

Approved for public release; distribution is unlimited.

Unclassified

security classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report Approved for public release; distribution is unlimited.
2b Declassification Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)

6a Name of Performing Organization Naval Postgraduate School	6b Office Symbol *(if applicable)* 35	7a Name of Monitoring Organization Naval Postgraduate School
6c Address *(city, state, and ZIP code)* Monterey, CA 93943-5000		7b Address *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a Name of Funding Sponsoring Organization	8b Office Symbol *(if applicable)*	9 Procurement Instrument Identification Number
8c Address *(city, state, and ZIP code)*		10 Source of Funding Numbers

Program Element No	Project No	Task No	Work Unit Accession No

11 Title *(Include security classification)* AUTOMATIC SATELLITE IMAGE NAVIGATION

12 Personal Author(s) Brian C. Spaulding

13a Type of Report Master's Thesis	13b Time Covered From To	14 Date of Report *(year, month, day)* September 1990	15 Page Count 94

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms *(continue on reverse if necessary and identify by block number)*
Field	Group	Subgroup	image navigation, binary correlation, automatic landmarking

19 Abstract *(continue on reverse if necessary and identify by block number)*

The satellite image navigation system for AVHRR (Advanced Very High Resolution Radiometer) imagery at the Naval Postgraduate School, referred to as Avian, has been modified from an operator interactive procedure to an automatic navigation procedure. The interactive procedure, based on operator identification of discrete landmarks, has been replaced with a procedure which utilizes the Defense Mapping Agency's (DMA) World Vector Shoreline (WVS) as a reference Binary shoreline images are created from the satellite images and correlated with the WVS reference shoreline using a sum of absolute differences matching technique. The correlation is performed using reference and search windows selected from full resolution sub-scenes of the WVS and satellite image.

Ten images were navigated with resulting accuracies of approximately 1.3 km. The resultant earth location was at least as accurate as the original Avian and did not depend upon expertise of the operator, as the original Avian procedure does. Thus, the automatic Avian procedure eliminated the subjectivity inherent in the interactive landmarking, while reducing the amount of expertise required to perform the navigation task. The automatic Avian procedure does not need a landmark atlas, so image navigation can be done globally since the World Vector Shoreline database has worldwide coverage. Currently, the only subjective and interactive step remaining is the placement of the reference and search windows to be correlated within sub-scenes. This step can be completed by an operator with no previous experience and not effect the accuracy of the navigation.

20 Distribution Availability of Abstract ☒ unclassified unlimited □ same as report □ DTIC users	21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual Professor C.H. Wash	22b Telephone *(include Area code)* (408) 646-2295	22c Office Symbol MR WX

DD FORM 1473,84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

Automatic Satellite Image Navigation

by

Brian C. Spaulding
Cartographer, Defense Mapping Agency
B.S., University of North Carolina at Wilmington, 1984

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN HYDROGRAPHIC SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author: [redacted]
Brian C. Spaulding

Approved by: [redacted]
Professor C.H. Wash, Thesis Advisor

[redacted]
Cdr. K.J. Schnebele, Second Reader

[redacted]
Curtis A. Collins, Chairman,
Department of Oceanography

ABSTRACT

The satellite image navigation system for AVHRR (Advanced Very High Resolution Radiometer) imagery at the Naval Postgraduate School, referred to as Avian, has been modified from an operator interactive procedure to an automatic navigation procedure. The interactive procedure, based on operator identification of discrete landmarks, has been replaced with a procedure which utilizes the Defense Mapping Agency's (DMA) World Vector Shoreline (WVS) as a reference. Binary shoreline images are created from the satellite images and correlated with the WVS reference shoreline using a sum of absolute differences matching technique. The correlation is performed using reference and search windows selected from full resolution sub-scenes of the WVS and satellite image.

Ten images were navigated with resulting accuracies of approximately 1.3 km. The resultant earth location was at least as accurate as the original Avian and did not depend upon expertise of the operator, as the original Avian procedure does. Thus, the automatic Avian procedure eliminated the subjectivity inherent in the interactive landmarking, while reducing the amount of expertise required to perform the navigation task. The automatic Avian procedure does not need a landmark atlas, so image navigation can be done globally since the World Vector Shoreline database has worldwide coverage. Currently, the only subjective and interactive step remaining is the placement of the reference and search windows to be correlated within sub-scenes. This step can be completed by an operator with no prior experience and not effect the accuracy of the navigation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to thank Professor C. H. Wash and Cdr. K. J. Schnebele for their guidance and constructive criticisms throughout my thesis research. Thanks also goes to the Defense Mapping Agency for sponsorship and to the Defense Mapping Agency and the Naval Postgraduate School for funding of the research. Special thanks to Craig Motell, who spent many hours working on the massive computer programming required to complete this work and also discussed all aspects of the research with me.

I. INTRODUCTION

Image registration or rectification is the process of eliminating errors or distortions in images so that corresponding objects in the two images correspond in size, shape and location. Satellite images are distorted due to Earth curvature, Earth rotation, spacecraft attitude, optical sensor limitations and small perturbations in satellite orbit (Emery and Ikeda, 1984). Registration can be between two images or between an image and a reference source, such as a map or chart. For satellite data to be efficiently utilized, it must be corrected and resampled to fit a desired map projection. This procedure has been termed image navigation, as the image is corrected and transformed to a known projection. This process locates each pixel at the appropriate geographic location (Emery et al., 1989). This is a fundamental requirement for the full utilization of satellite data, as distortions need to be removed from the images for the satellite data to be utilized as efficiently as possible.

The main goal of this thesis is to develop an automatic earth location algorithm to be used in conjunction with the current Naval Postgraduate School's navigation system, referred to as Avian. Avian utilizes ephemeris data and interactive landmarking (described later in this paper) to navigate satellite imagery. The interactive landmarking procedure is tedious and time consuming, and its accuracy is dependent upon the level of training and expertise of the operator. Here the Defense Mapping Agency's (DMA) digital World Vector Shoreline (WVS) is used which locates the shoreline of the world accurately (90% of all identifiable features can be located within 500 meters (Defense Mapping Agency, 1988)). This reference source is compared with the shoreline observed in NOAA's (National Oceanic and Atmospheric Administration) AVHRR (Advanced Very High Resolution Radiometer) imagery to replace the landmarking steps currently used. In automating the process the World Vector Shoreline is correlated with satellite data, using sections of the coastline to provide a "string of control points" rather than just using several manually selected ground control points (GCP's). This produces more timely and consistently navigated imagery from a procedure which can be applied globally.

Following the introduction a background on image navigation is given in Chapter 2. The WVS data is discussed in Chapter 3. The correlation procedure and methodology developed to automatically navigate the imagery is outlined in Chapter 4. Chapter

5 and 6 discuss the experimental design and the results of the experiment, respectively. These chapters are followed by conclusions and appendices. Appendix A discusses the original Avian procedure, Appendix B gives additional details of image navigation and Appendix C lists the computer code used to read the WVS data.

II. SURVEY OF IMAGE NAVIGATION

A. REVIEW OF REGISTRATION PROCEDURES

Many methods of registration have been applied to satellite imagery. Registration can be done by either the control point approach or the correlation approach (Davis and Kenue, 1978). Recent results using both methods are now reviewed.

1. Ground Control Point Registration

If the differences between images is any combination of translation, rotation and scaling the images can be registered by ground control points (GCPs) by determining the positions of a minimum of two corresponding points in the images (Goshtasby et al., 1986). A transformation function is obtained through a least squares analysis of the corresponding control points. First order (linear) transformations relate the control points and can be used for the images if parallel lines remain parallel in the second image. For non-parallel and other non-linear distortions, a higher order transformation must be used.

Registering images with translational differences has been examined by many authors (Barnea and Silverman, 1972; Jayroe et al., 1974; Pratt, 1974; Cordan and Patz, 1979; Leberl and Kropatsch, 1980; Eversole and Nasburg, 1983; Goshtasby et al., 1986). Images with translational differences can be registered by windowing techniques which use a number of windows in high variance areas of one image. The corresponding windows are located in the second image and the center of these windows are used as GCPs to determine the registration parameters (Barnea and Silverman, 1972). Rotational differences can be registered by lines and segments (Stockman et al., 1982) while scaling differences can be handled mathematically. Labowitz and Marvin (1986) found that between seven and eleven GCPs were needed in image registration of Landsat imagery. Orti (1981) developed optimal distributions of ground control points (at the four corners of the image and at areas at the right and left edges approximately one quarter of the distance between the corners) required to obtain a given average registration error. This keeps the number of points to be selected to a minimum, as manual determination of GCPs is tedious and time consuming.

Registration utilizing GCPs has been applied to both Landsat and AVHRR images. The tracking accuracy and the availability of high quality GCPs is much more limited for the AVHRR (Emery et al., 1989). As this thesis pertains to AVHRR imagery, the use of GCPs with AVHRR will be examined.

There are basically two approaches to GCP image navigation of AVHRR imagery (Emery and Ikeda, 1984; Emery et al., 1989). One method (the circular orbit method) assumes a circular spacecraft orbit based on initial ephemeris data and relies on known GCPs to correct for errors in Earth shape, scan geometry, satellite orbit and satellite attitude, to provide improvement in image to map registration. The second method (the ephemeris data method) relies on highly accurate satellite ephemeris data and only uses GCPs to correct for timing errors and satellite attitude angles. Emery and Ikeda (1984) state that seven GCPs are needed with the circular orbit method to achieve similar accuracy as the ephemeris data method using one GCP. Both methods are examined in more detail in sections to follow.

2. Correlation Registration

Registration can also be accomplished by correlation techniques (Jayroe et al., 1974; Cordan and Patz, 1979; Leberl and Kropatsch, 1980; Sullivan and Martin, 1981; Sun, 1981; Anuta and Davallou, 1982; Crombie, 1983; Henderson et al., 1985; Anuta and McGillem, 1986). The two images are decomposed into a number of overlapping sub-scenes. The local shift parameters are obtained by a similarity detection algorithm after cross-correlation of a sub-scene from the reference image with a sub-scene from the other image. The shift parameters are obtained for all of the sub-scenes and a least squares analysis is used to compute the best fit linear transformation (Davis and Kenue, 1978).

Binary gradient images have also been utilized for image registration (Jayroe et al., 1974; Cordan and Patz, 1979). These images consist of only two pixel values. For example, when using grey level values. the minimum and maximum grey level value may be the only grey level values contained at each pixel. The binary images are obtained by thresholding the gradient images with respect to some threshold value and then the binary gradient pictures are registered by translation, rotation and scaling of sub-scenes (Davis and Kenue, 1978). Anuta and McGillem (1986) discuss three correlation methods of registration. These are direct correlation of gray tone image patches, correlation of edge images and intersecting line segments (which they termed the parameter space method). The parameter space method is based on the assumption that the images contain linear features which can be described by sets of parametrically defined line segments. These can be matched with the corresponding parameters of another image and the registration transformation determined. These three techniques were found to be equivalent at high signal-to-noise ratios, but the parameter space method proved to be more successful at low signal-to-noise ratios.

Edge images have been used for image navigation. Edge detection is an image segmentation method based on the discontinuity of gray levels or textures at the boundary between objects. An edge separates two regions of relatively uniform but different gray levels or textures. Edges are obtained by edge enhancement techniques, such as high pass filtering, Laplacian operators and gradient operators (Moik, 1980). Nack (1977) applied edge image correlation techniques to Landsat-1 multispectral digital image data. Henderson et al. (1985) examined edge- and shape-guided correlation of control point areas and detailed ways of using edge descriptors in various registering procedures. Novak (1978) discussed correlation algorithms using edge detection for radar images. Gupta and Wintz (1975) developed a boundary finding algorithm for locating gray level and or texture edges based on hypothesis testing. Wong and Hall (1979) examined several scene matching techniques, including scene matching with edge features which utilizes a correlation coefficient in its process.

A specific utilization of a correlation technique is template matching, which is applied in the procedure developed in this paper. Template matching is the process of locating the position of a sub-image inside a larger image. The sub-image is called the template (or the window area or the reference window area) and the larger is called the search area (Hall, 1979; Moik, 1980; Eversole and Nasburg, 1983; Goshtasby et al., 1984). Examples of the templates on the reference image and the corresponding search areas on the search image are shown in Figure 1.

The matching process involves shifting the template over the search area and computing the similarity between the template and the window in the search area over which the template lies. Then the shift position where the largest similarity measure is obtained can be determined. Similarity measures that have been used with this technique are the sum of absolute differences (Vanderburg and Rosenfeld, 1977; Hord, 1982) and the cross correlation coefficient (Moik, 1980; Hord, 1982; Goshtasby et al., 1984). Goshtasby et al. (1984) state that the sum of absolute differences is computationally fast, while the correlation coefficient measure is more accurate, but computationally slow. Hord (1982) indicates that a major advantage of the sum of absolute differences as a mismatch measure is that this measure grows monotonically with the number of points whose absolute differences have been added into the sum. This means that when looking for a point of best match in a given region, the process can be stopped when the sum of a given position has grown larger than the smallest sum obtained up to that point

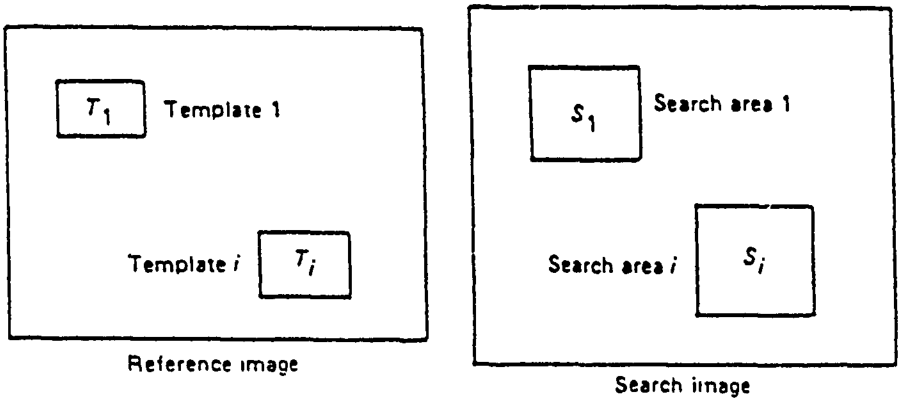


Fig. 1. Templates and corresponding search areas (Moik, 1980)

from previously evaluated positions. Hord also points out that an absolute threshold can be used in conjunction with this technique to further increase the speed of the procedure. Absolute thresholding involves setting a value that is the cutoff value for the summing procedure. Then at any point, if this value is reached, the process will be stopped and move to the next point.

To increase the speed of the search process for either similarity measure, a two-stage template technique can be used. This method first uses a subtemplate to determine the possible position for a match by determining the shift position which results in a similarity measure above some threshold value. This eliminates investing time on positions which show no evidence for a match. After this is accomplished, the entire template can be used over the good match areas to determine the best match position. Two-stage template matching can also be accomplished by first using a reduced resolution template as the subtemplate in the first stage (Rosenfeld and Vanderburg, 1977). Two-stage template matching techniques have resulted in reduced computation time for both sum of absolute differences (Rosenfeld and Vanderburg, 1977) and for the cross correlation coefficient method (Goshtasby et al., 1984).

3. Summary of image navigation

Currently, GCPs are being used in conjunction with ephemeris data to navigate satellite imagery. The procedure and type of ephemeris data provided determine the number of GCPs needed to obtain accurately navigated imagery. Additional details of this are provided in the following section on AVHRR image navigation. Correlation techniques have been used by several authors to automatically identify GCPs to be used in the registration process (Nack, 1977; Ng, 1977; Davis and Kenue, 1978; Ho and Asem, 1984). This eliminates human interaction at the GCP acquisition step, thus reducing the introduction of human subjectivity and the resulting variability of the navigated image which will occur with different operators. These automated techniques is discussed in greater detail in later sections.

B. AVHRR IMAGE NAVIGATION

As mentioned above, there are two approaches to image navigation of AVHRR imagery with GCPs. These methods are referred to as the circular orbit method and the ephemeris data method. Both methods make use of ephemeris data and have been used successfully to accurately navigate satellite imagery. Ephemeris data are the information that describes the location of a satellite in the geocentric reference frame.

1. Circular Orbit Method

The circular orbit method assumes a circular spacecraft orbit based on only approximate orbital parameters from initial ephemeris data provided by NOAA. Further corrections are carried out by the use of GCPs (Emery and Ikeda, 1984). These correct for errors in Earth shape, scan geometry, satellite orbit and satellite attitude. This method is more time consuming than the ephemeris data method as more landmarks are needed to obtain similar accuracies. This increases the amount of time spent with an operator selecting GCPs, unless pre-selected points are always used. There is also the problem of cloud cover obscuring possible landmarks. Additionally, there may not be much land present in the image and therefore, few landmarking possibilities.

Legeckis and Pritchard (1976) used an algorithm with a circular orbit and a spherical earth, only correcting the geometric distortions due to Earth curvature, Earth rotation and spacecraft roll and obtained navigated imagery (NOAA-4 data) with an accuracy of approximately 5 km. NOAA/NESDIS (National Environmental Satellite and Information Service) and CMS (Centre Meteroligie Spatiale) both use automated versions of this procedure, using a circular orbit and spherical Earth and obtain accuracies to within 5 km (Emery et al., 1989). Emery and Ikeda (1984) used initial ephemeris data (assuming a circular orbit) with seven GCPs to obtain accuracies up to 1.5 km, while Rauste and Kuittinen (1985) navigated NOAA images with the root mean square error of the least squares adjustment was between 3.7 (approximately 3.4 pixels) and 6.1 km. Ho and Asem (1986) developed a model assuming a spherical Earth and circular orbit, but took into account the rotation and oblateness of the Earth, following Duck and King (1983) and achieved accuracies as high as approximately 3 km while using only one GCP. The single GCP was used to correct for variations in the satellite attitude and inclination angle. Ho and Asem (1984) also developed an automated navigation process by using a maximum correlation technique to identify and locate GCPs from a set of previously well defined GCPs. Emery et al. (1989) discusses the correction calculation of satellite altitude in detail, using one GCP. This correction adjusts for image distortions due to the oblateness of the earth and the ellipticity of the satellite orbit. Emery and Ikeda (1984) used seven GCPs to make this correction, using a least-squares fit to allow more GCPs to be used. Scan skew needs to be adjusted for as well, due to the fact that pixels are larger and do not change much away from nadir (Brush, 1988). However, pixels near the sub-satellite point change size rapidly. This can be corrected for by linearizing the scan geometry. Equations for linearizing the scan geometry are reviewed by Emery et al. (1989).

2. Ephemeris Data Method

Image navigation can be carried out utilizing an elliptical model if there is access to regularly updated ephemeris data. This ephemeris data is highly accurate and is supplied daily by U.S. Navy tracking stations. Several papers have been published, each outlining navigation models relying on this type of ephemeris data (Brush, 1985, 1988; Emery and Ikeda,1984; Brunel and Marsouin, 1987). Brush (1988) and Brunel and Marsouin (1987) recommend a fully elliptical orbital model of relatively low eccentricity (Keplerian) where orbital parameters are specified by a set of accurate orbital elements computed from the tracking of the satellites (Emery et al., 1989). The orbital parameters needed for image navigation (Emery and Ikeda, 1984) are:

ϕ_o	the longitude of equatorial passage for the orbit of interest
c	the local value of satellite angular velocity
H	the local value of satellite altitude
t_o	the equivalent equatorial passage time
M_o	the mean anomaly
T_o	Epoch, midnight GMT
ω_o	argument of the perigee at Epoch
M_1	the mean orbital motion
M_2	the orbital decay rate
λ_o	the longitude of equatorial passage at Epoch
e_o	the eccentricity
i	the inclination

The equivalent equatorial passage time (t_o) is extrapolated using c along with ephemeris values of the mean anomaly. These orbital parameters are calculated from the ephemeris data, details of these calculations are given by Emery and Ikeda (1984) and in the summary article by Emery et al. (1989).

These parameters can then be used to calculate the line and pixel locations that match a selected geographic map. These corrected pixel locations are calculated in x (position along the scan line) and y (change in scan line) to allow intercomparison between corrected images. Ho and Asem (1986) call this inverse image referencing. Direct image referencing is the process of distorting the geographic grid to match the satellite image projection, thus making it difficult to compare different images.

Emery et al. (1989) also discuss a method of interpolation and remapping to decrease the time required for the image navigation. Several resampling techniques also are described by Kashef and Sawchuk (1983). Several authors (Brush, 1985, 1988; Brunel and Marsouin, 1987) use ephemeris methods which do not require GCPs, resulting in accuracies ranging from 1.6 to 10.4 km. Brunel and Marsouin used clock error values obtained from NOAA/NESDIS to update the time, while Brush utilized a "nudge" to align the coastline boundary to account for the timing error. Emery and Ikeda (1984) state that after their image correcting procedure there are small deviations in geography resulting from short-term variations in orbital parameters and fluctuations in satellite attitude (roll, pitch and yaw). Changes in spacecraft attitude affect the orientation of the radiometer, thus distorting the image. To correct for these errors Emery and Ikeda used seven GCPs using a least squares procedure, to determine how each pixel needed to be shifted. However, the offsets corrected by this procedure were primarily in the meridional direction along the path of the satellite. These meridional offsets are primarily due to errors in the recorded time of image data collection and are all of similar size and can be largely corrected for by using only one ground control point (Emery and Ikeda, 1984). With this procedure, the authors obtained maximum accuracies up to 1.5 km.

Emery et al. (1989) navigated images using the elliptical model to locate the center point of the images and then applied the circular orbit approximation to a spherical Earth. Then the offset along the center of the nadir track (due to the lack of accurate time at the receiving station) was corrected for with a "nudge" along the trackline to bring the coastal boundary into agreement with the image. They also navigated the same images with a fully elliptical model for the entire calculation, with the pixel locations between these two procedures having an average offset of about three pixels (approximately 3 km). The authors report navigation accuracies up to 1.5 km are possible with these techniques.

3. Comparison of ephemeris data and circular orbit methods

The ephemeris data method is much faster than the circular orbit method as there is no operator interaction required to compute the geometric corrections and projection resampling needed for the navigation (Emery et al., 1989). However, one of the largest sources of error is inaccurate timing at the ground station and/or on the satellite. The drift error on the timing devices on TIROS-N satellites is approximately 70 ms per day and if there is not an accurate tii ıe standard to know exactly when the data was collected there will be errors along the satellite track. This along track shift can be corrected with a "nudge". This nudge moves the entire image up or down along the nadir trackline until the geographic features are lined up best and often a single GCP is used to determine this along-track nudge (Emery et al., 1989). So the ephemeris data routine has options for corrections to match known GCP locations to obtain information for this nudge. If the errors attributed to the attitude of the satellite are to be corrected, three GCPs are needed. Emery et al. (1989) suggests this can be neglected, however, and still achieve accuracies on the order of one km, as the attitude of the polar orbiters is controlled to better than 0.1 degrees in roll, pitch and yaw axes.

Both of these methods can navigate images to similar accuracies, however, the circular orbit method needs to use more GCPs to obtain accuracies similar to the ephemeris data method. This increases the amount of time and human interaction needed to navigate the images with the circular orbit method.

C. NPS IMAGE NAVIGATION SYSTEM (AVIAN)

The Naval Postgraduate School navigates digital imagery data with an ephemeris data system (Avian) which utilizes highly accurate ephemeris data and an interactive process to select landmarks. The following description of the system is taken from Bethke (1988).

The digital imagery data used by this system are provided by NOAA ground receiver stations, such as Scripps Institution of Oceanography, and are referred to a HRPT (High Resolution Picture Transmission) data. These data are obtained from the Advanced Very High Resolution Radiometer (AVHRR) sensor flown on NOAA polar orbiting satellites. The ephemeris data on the digital tapes represent the orbital elements of the satellite at 0000Z Greenwich Mean Time (GMT). However, the majority of the imagery provided on the HRPT tapes is generated at a time T+0000Z GMT. From the time

0000Z to T+0000Z, the satellite has moved west relative to the earth in its orbit (approximately 15 degrees per hour). Therefore, the satellite's position, from the ephemeris data, must be updated to the time T, so that the satellite's position agrees with the digital data provided on the HRPT tape.

The satellite's position is updated from 0000Z GMT using the navigation model which is designed to take the time variance of the orbital elements and the perturbative effects of various forces acting upon the satellite into consideration. Given the updated satellite position, the sub-satellite point can be determined. The satellite's ascending/descending node is then calculated using the updated ephemeris data and known spherical geometry relationships.

Once the subsatellite and the nodal points are calculated, the positions of the landmarks relative to the geocentric Earth reference system can be found. The satellite imagery can then be "mapped" or navigated to the Earth using the reference landmarks. The Avian procedure is reviewed in Appendix A and the theoretical background of the navigation process is given in Appendix B.

The Avian model used at the Naval Postgraduate School has provided accuracies between approximately 2.5 and 5 km when using NOAA AVHRR data of 1.1 km resolution, as reported by Bethke (1988). Results of these tests and other navigated imagery on this system indicates that the level of training of the operator will have a great effect on the resulting accuracies of the navigation. St. Pierre (1988) reported that increasing the level of the system's operator training would probably increase accuracy of this system to the individual pixel level. The author of this thesis has navigated imagery to the pixel level. This illustrates one of the primary reasons for automating the landmarking system, as there is a need to remove the interactive, time consuming and somewhat subjective landmarking procedure. An automated procedure could reduce processing time so that the navigated images could be obtained in nearly "real time" and there would also be the added benefit of removing errors that are introduced into the system by inexperienced or less trained operators. This would remove training time for the landmarking procedure as well as producing more consistently navigated imagery anywhere on the globe.

D. SUMMARY OF NAVIGATION ACCURACIES

The accuracy of satellite image navigation schemes varies depending on the method chosen and on the implementation of that method (i.e., what number of GCPs are to be used, what is the distribution of the GCPs, the amount of human interaction and subjectivity introduced, etc.). Table 1 outlines the methods discussed in this chapter and shows the approximate accuracy of each method. Table 1 indicates the current accuracies obtainable and illustrates basic accuracies which the method described in this paper can be compared.

Table 1. APPROXIMATE ACCURACIES OF AVHRR IMAGE NAVIGATION SCHEMES

Author	Method	Accuracy (in km)
Legeckis and Pritchard 1976	Circular Orbit, 0 GCPs	5
Emery and Ikeda 1984	Circular Orbit, 7 GCPs	> 1.5
Rauste and Kuittinen 1985	Circular Orbit, ? GCPs	3.7 to 6.1
Ho and Asem 1986	Circular Orbit, 1 GCP	3
Brush 1988	Ephemeris Data, nudge	2 to 3
Brunel and Marsouin 1987	Ephemeris Data, 0 GCPs	1.6 to 10.4
Emery and Ikeda 1984	Ephemeris Data, 1 GCP	> 1.5
Bethke 1988	Ephemeris Data, 2-16 GCPs	2 to 5

Note that in Table 1, the method by Rauste and Kuittinen does not list the number of GCPs used. Their method utilized GCPs but the number of GCPs used was not given in their paper.

III. WORLD VECTOR SHORELINE

The goal of this thesis is to use the Defense Mapping Agency's World Vector Shoreline (WVS) as a reference base to earth locate satellite imagery. WVS is a digital data file containing the shorelines, international boundaries and country names of the world. The absolute horizontal accuracy requirement for WVS data is that 90% of all identifiable shoreline features be located within 500 meters circular error of their true geographic positions with respect to the preferred datum (World Geodetic System). The shoreline data is produced from cartographic or imagery source material. Boundaries and Mean High Water shorelines in the WVS data set are referenced to the World Geodetic System (WGS) datum and ellipsoid.

The WVS format uses a chain-node (or chain-code or chain encoded) data structure to minimize redundant data storage. Each point of intersection is explicitly defined to eliminate gaps or overlaps between segments of adjacent features. These points are referred to as nodes (endpoints of segments) and the order of points within a segment defines the direction, which allows identification of areas to the left and right of the segment (Defense Mapping Agency, 1988).

Chain-codes are more efficient than sequences of points that are represented by X-Y coordinates (Pavlidis, 1982). Rather than digitizing an entire picture in the conventional way, a mesh is placed over the analog picture and the vertices closest to where the curve crosses the lines on the mesh are identified. These vertices are taken to represent the curve and the vertices can then be encoded in an octal (eight direction) sequence by giving the direction from one vertex to the next (Duda and Hart, 1973). WVS chain-node utilizes a common chain-code that uses eight directions, which indicate which di rection the feature is entering the cell of interest (Fig. 2).

The chain-node method differs greatly from a sequential storage method. In the sequential storage method, individual features, as shown in Fig. 3, would be treated as separate polygons, digitized independently and stored as a string of X-Y coordinates. The common boundary between two abutting features would be digitized twice and the two versions may not coincide, which could create gaps and overlaps along these boundaries.

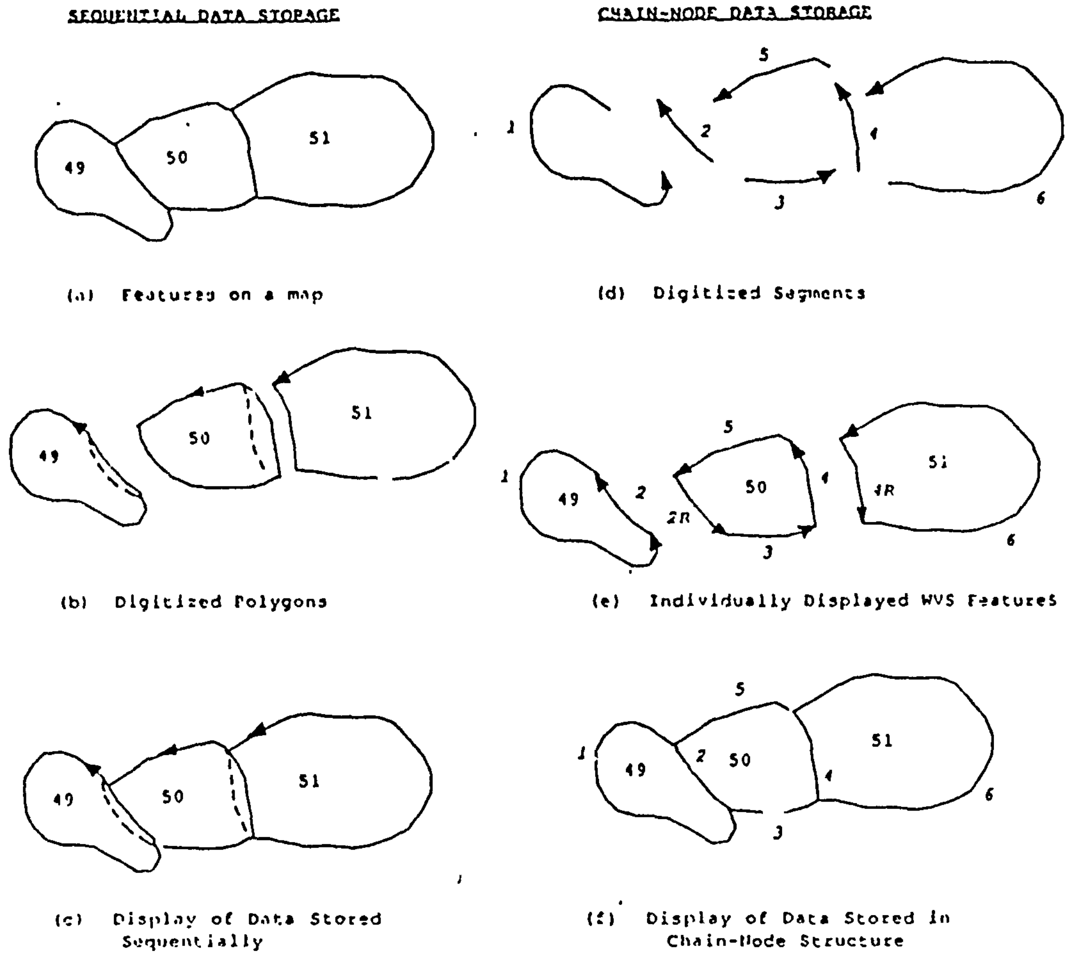
In the chain-node method any common feature boundary is only digitized once, so that the features will abut precisely and that boundary will not be stored redundantly.

3	4	5
2	cell of interest	6
1	8	7

Fig. 2. Illustration of Edge codes (Defense Mapping Agency, 1988)

The coordinates of each segment are stored with a unique alphanumeric ID so that the segment can be displayed separately or with other segments as a feature.

Using WVS as a reference source has several advantage over other reference sources, such as charts and maps. The WVS is a very accurate source of the shoreline and covers the entire world. Direction of the vertices allows easy determination of the areas to the left and right of the segment, so that in unfamiliar areas, land and water areas are distinguishable. The WVS database is extremely useful in image navigation, as it is readily available with worldwide coverage on a consistent geocentric datum and, as is discussed later, is easily presented is a binary image format. This makes it ideal for matching techniques involving coastlines obtained from satellite images.



(a) Features on a map

(d) Digitized Segments

(b) Digitized Polygons

(e) Individually Displayed WVS Features

(c) Display of Data Stored Sequentially

(f) Display of Data Stored in Chain-Node Structure

Sequential vs. Chain-Node Methods of Cartographic Data Storage. Arrows indicate the ordering of coo-dinates within a segment: solid lines represent digitized outlines; dashed lines represent true boundaries.

Fig. 3. Illustration of Sequential vs. Chain Node data storage (Defense Mapping Agency, 1988)

IV. AUTOMATING AVIAN

A. OVERVIEW OF CORRELATION PROCEDURE

A correlation procedure has been developed and tested that utilizes the World Vector Shoreline (WVS) data and the shoreline in satellite images to automatically navigate AVHRR images. Leberl and Kropatsch (1980) state that for automatically merging maps and satellite data, there is no limit to the number of possible pattern recognition techniques that can be used. Several authors (Ng, 1977; Davis and Kenue, 1978; Ho and Asem, 1984) have discussed automatic GCP acquisition. Ho and Asem's technique used a library of gradient values for comparison to particular landmarks. This is a disadvantage as there would be a great deal of time spent to develop this library and there could be situations where imagery could not be navigated if the known GCPs were obstructed by cloud cover. The papers by Ng and by Davis and Kenue both discuss correlation approaches to obtain GCPs. The procedure discussed by Davis and Kenue is of particular interest as it is similar to the procedure used in this paper. They used binary gradient images in combination with a correlation technique. Windows were chosen within the binary images. A correlation equation would then be applied over the windows. A GCP would be chosen where the largest correlation value was obtained. This technique was used to register between two Landsat images of the same area, taken from different angles.

The most commonly used matching technique is the correlation coefficient (or normalized cross correlation) (Hord, 1983). The disadvantage of this technique is that it is relatively slow, as it involves a large number of multiplications. The technique chosen for this experiment is the sum of absolute differences (SAD). This technique is computationally more efficient than many of the cross-correlation techniques as it not only does not require multiplications, it also requires fewer arithmetic operations than cross correlation techniques. This procedure has been applied with binary shoreline images. The two images that are matched consist of only two gray level values. These are either zero (for zero gray level) or 255 (the maximum gray level). The maximum gray level will comprise the shoreline and the zero values the background. To apply the SAD procedure, each pixel value on one image is subtracted from the corresponding pixel value on the second image, then the absolute value is taken and summed over that entire image. If there is a perfect match, the SAD value is zero. By shifting the 'reference window' from the reference image (WVS image) over a 'search window' from the satellite image, the best matching position or lowest SAD value can be found. Hord (1982) states that one advantage of this method is that once the sum at a given position is larger than the smallest sum obtained before that point, the process can be stopped for that position, as it will not be a best match position. This can greatly reduce the number of arithmetic operations performed and thus save time.

A flow chart (Fig. 4) illustrates key steps that were used to implement the correlation procedure together with Avian capabilities to produce automatically navigated imagery. Most of the steps described for the original Avian procedure (Appendix A) can still be utilized with the automatic Avian procedure. This new method eliminates the GCP selection procedure (Pickem) and modifies the procedure (Twiddle) which obtains the timing error and the satellite attitude errors. All of the remaining steps in Avian are still viable.

B. METHODOLOGY

1. Glean and overview

The first steps in this procedure are identical to the start of the original Avian navigation procedure as described in Appendix A. Glean is the process of reading all of the satellite image data from the AVHRR pass. Overview produces an image on a display monitor of the entire satellite pass at reduced resolution. This needs to be completed before sub-scenes of the image can be produced.

2. Paint a sub-scene

After an overview has been created, a sub-scene of the image is 'painted' utilizing the existing [illegible] procedure of Avian. The starting line and pixel number needs to be recorded by the operator, as this will be needed to run the automatic matching procedure. These sub-scenes can be produced for several different channels. Figures 5 and 6 are sub-scenes of an AVHRR pass of Southern California and Baja. Through experimentation, it was found that a sub-scene produced using a ratio of albedo in channel 1 to channel 2 worked well. This produced sub-scenes where the land-water interface was sharply defined, and tended to reduce the effect of clouds over this boundary and over the water just off the coast (Fig. 6). Channel 2 also worked well, producing sharp coastlines but clouds reflect very brightly in this channel and could cover the coastline, making sections of the image unusable (Fig. 5). Figure 5 shows cloud cover off the coast (in the channel 2 image) and Figure 6 illustrates reduced effects of cloud cover when the albedo ratio technique is used.

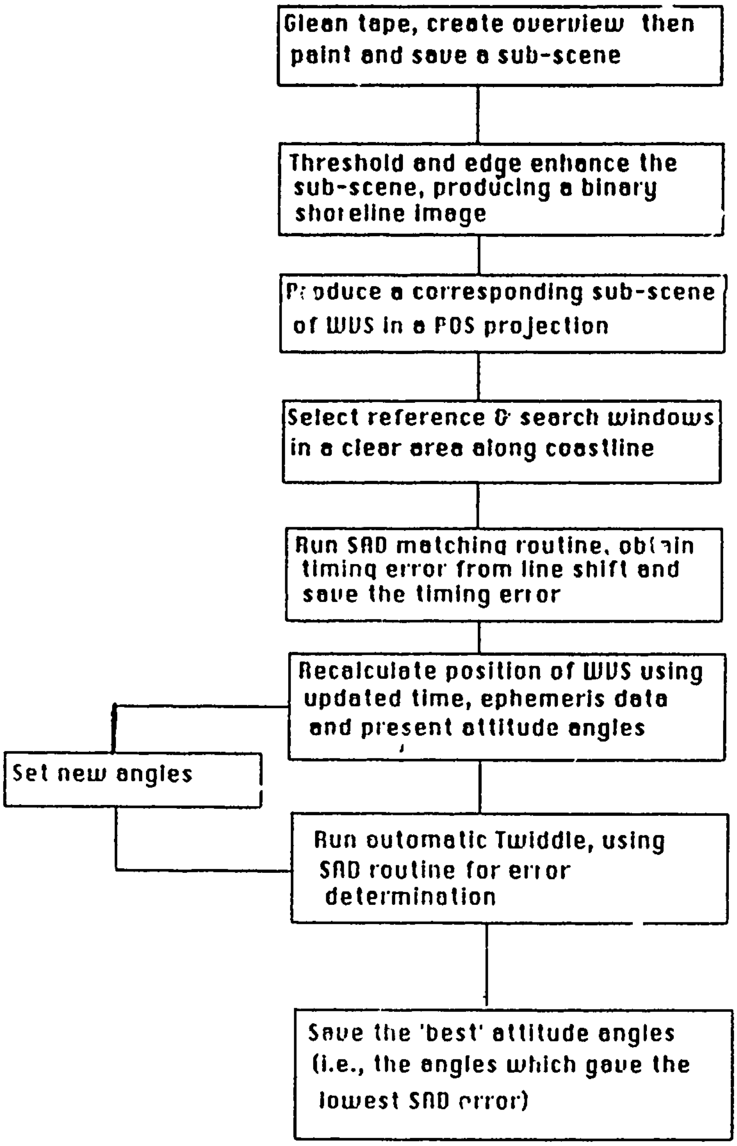

Glean tape, create overview then paint and save a sub-scene
Threshold and edge enhance the sub-scene, producing a binary shoreline image
Produce a corresponding sub-scene of WUS in a POS projection
Select reference & search windows in a clear area along coastline
Run SAD matching routine, obtain timing error from line shift and save the timing error
Recalculate position of WUS using updated time, ephemeris data and present attitude angles
Set new angles
Run automatic Twiddle, using SAD routine for error determination
Save the 'best' attitude angles (i.e., the angles which gave the lowest SAD error)

Fig. 4. Automatic Avian procedure

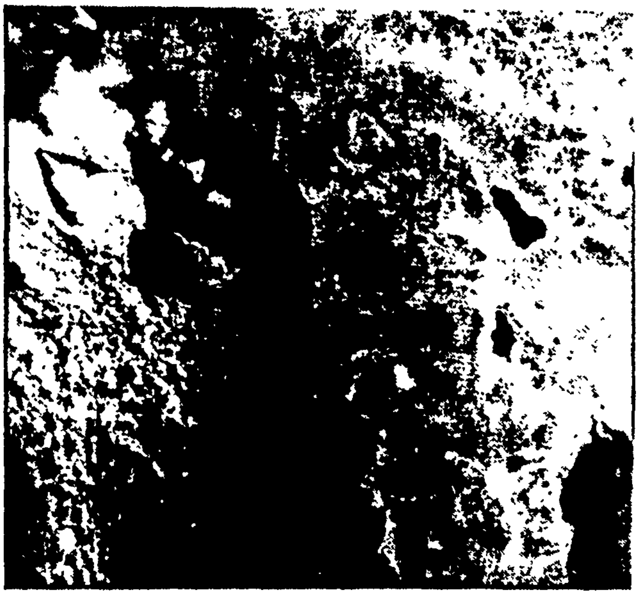


Fig. 5. Sub-scene from ch. 2

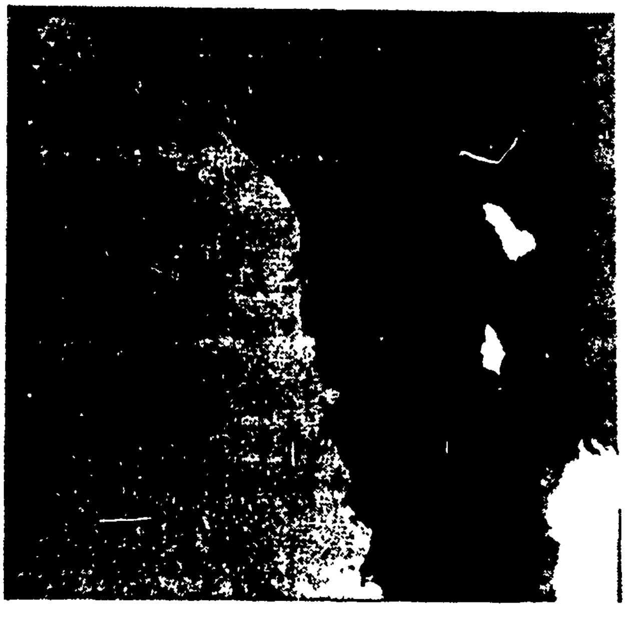


Fig. 6. Sub-scene from ratio of albedo ch.1/ch.2

3. **Produce a binary satellite sub-scene**

The sub-scene created by Paint undergoes several preprocessing techniques to prepare the image for the SAD matching. The sub-scene is thresholded in order to produce an image that contains only two gray level values, that is to produce a binary image. These values are zero (for the water) and 255 (for land). A value between zero and 255 is chosen that produces a sharp contrast at the land/water interface. In this study a brightness count of 45 (approximately 13% reflectance) has produced sharp coastlines for many of the images tested. All of the pixels that contain gray level values lower than the cutoff value are assigned a value of zero. All pixels containing gray levels of 45 or higher are assigned values of 255. For a sub-scene created in a visible channel, the land areas are white and the water areas are black.

4. Edge enhance the sub-scene

When the sub-scene has been converted into a binary image the next step is to edge enhance the sub-scene to produce an image consisting of a coastline only (Figs. 8 and 9). There are many operators used for edge enhancement. The Robert's gradient operator (or the Robert's cross operator) was chosen and resulted with binary images that contained distinct coastlines. Leberl and Kropatsch (1980) recommended the Robert's gradient operator as it gave high performance with modest computing requirements. This operator is represented mathematically in the following equation (1).

$$R(i,j) = \sqrt{\left[(g(i,j) - g(i+1,j+1))^2 + (g(i,j+1) - g(i+1,j))^2\right]} \qquad (1)$$

where

R(i,j)	value of Robert's gradient operator at a cell or pixel
g(i,j)	a particular cell or pixel
i	line number (or row number)
j	pixel number (or column number)

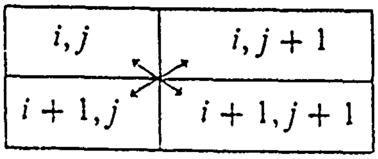


Fig. 7. Robert's gradient operator (Duda and Hart, 1973)

Figure 7 illustrates that diagonally adjacent pixels are utilized within the procedure. This results in a directional derivative in each direction that is approximated by simply subtracting adjacent elements. This step of the procedure yields a sub-scene with a binary representation of the coastline only, with a background of zero gray level values.

By applying the thresholding and edge enhancement techniques to Figures 5 and 6 (which are the sub-scenes created with channel 2 and the ratio of albedo of channel 1 to channel 2, respectively) binary shoreline images of both sub-scenes are created (Figs. 8 and 9). Both sub-scenes contain areas of unobstructed shoreline which could be utilized with correlation or matching procedures. For this study channel 2 was chosen, however, as it took the Paint procedure twice as long to produce a sub-scene using the ratio of albedo of channel 1 to channel 2 as compared to using channel 2 alone. Also, only small 'windows' are used to match areas on the image, so clear coastline areas can be chosen easily, eliminating the need to use the channel 1-2 ratio. The ratio of albedo method may be of more use if the window size is increased and the need to avoid clouds is greater. Figures 5, 6, 8 and 9 illustrate the effects of using the ratio of albedo of channel 1 to channel 2 and of using channel 2 alone. Figures 6 and 9 were produced using the channel 1-2 ratio method and Figures 5 and 8 used channel 2. When the binary shoreline images (Figs. 8 and 9) are examined, it can be seen that the channel 1-2 ratio method greatly reduced the effect of light cloud cover just off of the coast. The channel 2 image had a greater amount of 'noise' or clutter offshore that could adversely effect matching results if included in window areas. This noise is produced by clouds which appear as non-coastal edges within the binary image (Fig. 8).

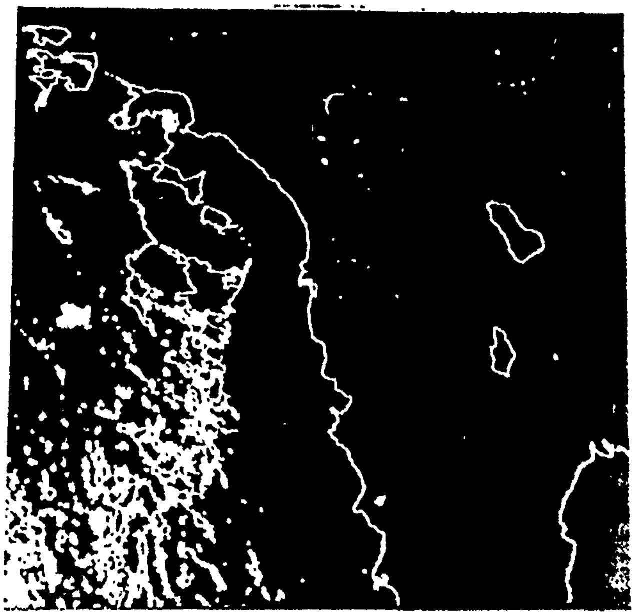


Fig. 8. Edge enhanced sub-scene from ch. 2

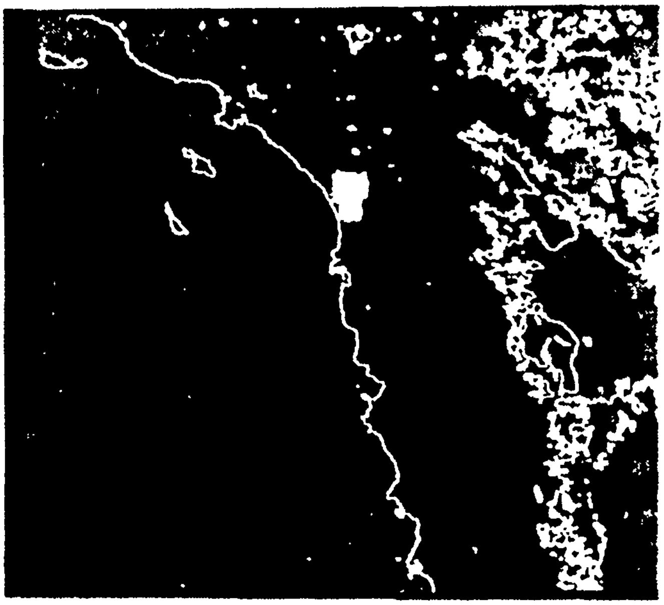


Fig. 9. Edge enhanced ratio of albedo sub-scene

5. Obtain WVS shoreline sub-scene

The next step is obtaining a corresponding image of WVS data, within the same projection as the satellite sub-scene. The first step in generating this image is to obtain the set of latitude/longitude WVS points. The program to read and select the points from the WVS tape is given in Appendix C. The shoreline is then projected into the POS (Polar Orbiter Satellite) projection by utilizing the ephemeris data that is provided on each HRPT tape. The latitude/longitude boundary from the satellite sub-scene is used, so this results in a binary shoreline image in the same projection and scale as the satellite sub-scene. An example of WVS in a POS projection is illustrated by Figure 10.

6. Determine windows in sub-scenes

Selecting corresponding windows in each sub-scene is the next step in this procedure. The WVS image is considered the reference image (in the literature this has also been termed the template image) and the satellite sub-scene is referred to as the search image. Each of these images will have a 'window' chosen within them where the actual SAD process will be calculated (Fig. 11). The reference window is smaller than the search window. The reference window must be smaller because the reference window is moved throughout the search window in order to determine the shift which best aligns the two windows. This shift is determined by how many lines and pixels the search window must be translated to 'match' the reference window. In order to determine what size the windows should be in relation to each other (to insure a match), some a priori information is needed. This information was obtained by performing a forward navigation using no landmarks (which means that the along-track error and the satellite attitude angles were not corrected). This allows the accuracy of the images to be determined before any correction process took place, thus allowing the relative size of the windows to be determined. That is, the approximate amount of offset between the two images that are being matched needs to be known. This enables the sizes of the windows to be determined so that the features that are being correlated are present in each window. The average offset obtained from ten navigations was approximately 12 km. Using this information (Table 2, in Chapter 6), the reference window size is chosen to be 32 X 32 pixels and the search window to be 64 X 64 pixels. The reference window is always be centered within the search window.

7. Correlate the sub-scenes

This procedure determines the displacement between the reference and search windows, thus resulting with the number of lines and pixels (rows and columns) the search window needs to be translated to align with the reference window. Figure 12

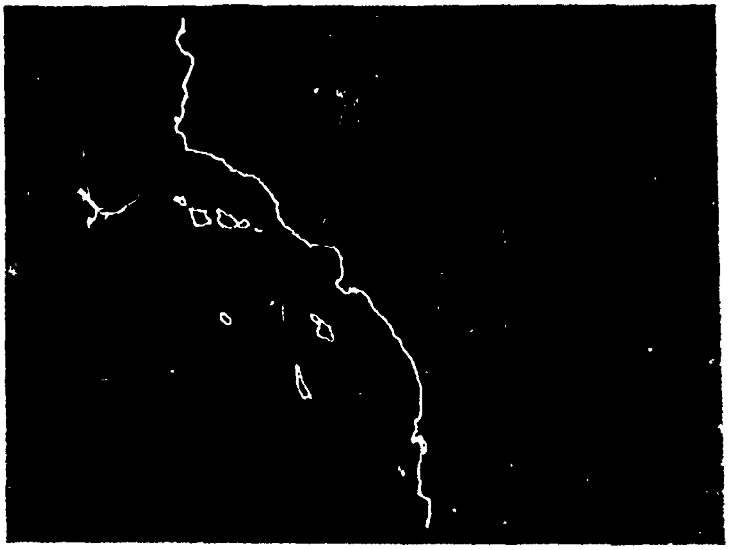


Fig. 10. A WVS sub-scene in POS projection

shows the entire reference and satellite sub-scenes overlaid before the along track and attitude angle errors have been corrected. The matching technique chosen for this correlation scheme is the sum of absolute differences (SAD). The SAD technique is given by:

$$SAD = \sum\sum | s(i,j) - r(i+u,j+u) | \tag{2}$$

The window values are represented by s and r, with i and j representing the line (row) and pixel (column) numbers and u being a shift amount. The result is zero when s and r are identical.

The reference window is originally centered with the search window. As this matching procedure is run, the reference window is 'moved' to align within the upper left corner of the search window. The SAD values (line and pixel shifts) are calculated at that position. The reference window is then moved one pixel at a time across the search window, with the SAD values being calculated at each position. After the entire 'row'

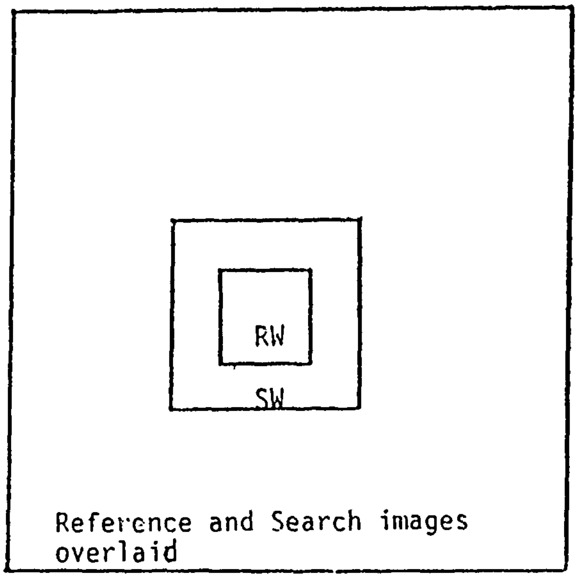


Fig. 11. Examples of reference and search windows

has been completed, the reference window is moved down one line and is sequentially moved across that row, calculating the SAD values at each position. This continues until the reference window has been moved throughout the entire search window, with the final position being the lower right corner of the search window. The translation position of the reference window that has the lowest SAD values is chosen as the shift amount that needs to be applied to the search image to align it with the reference image. Figure 13 illustrates the positioning of the reference window within the search window at its first and last computational position.

Obtaining this shift amount allows calculation of the along track error due to the timing error. Six lines are scanned per second with the AVHRR radiometer. Since this routine calculates the line and pixel shift, the line value can be used to determine the along track shift needed to correct for the timing error. Within the original Twiddle procedure, an average of the line offset of the landmarks chosen was used for this correction. Using the value obtained by matching one window area is actually similar to averaging many GCPs, corresponding to the number of pixel values for the shoreline of that window, since the shift obtained is an average shift for that window.

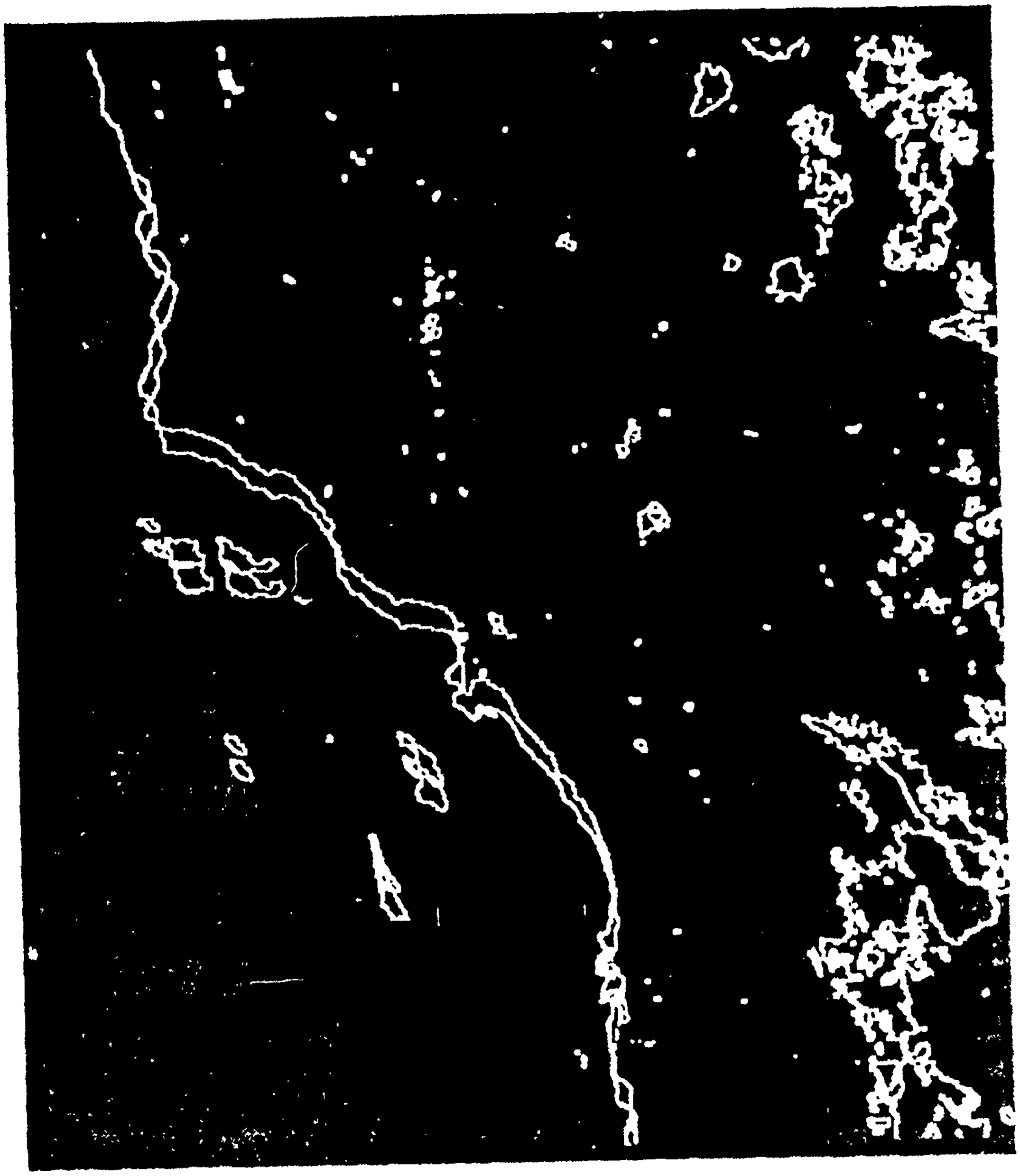


Fig. 12. Binary reference and satellite sub-scenes overlaid

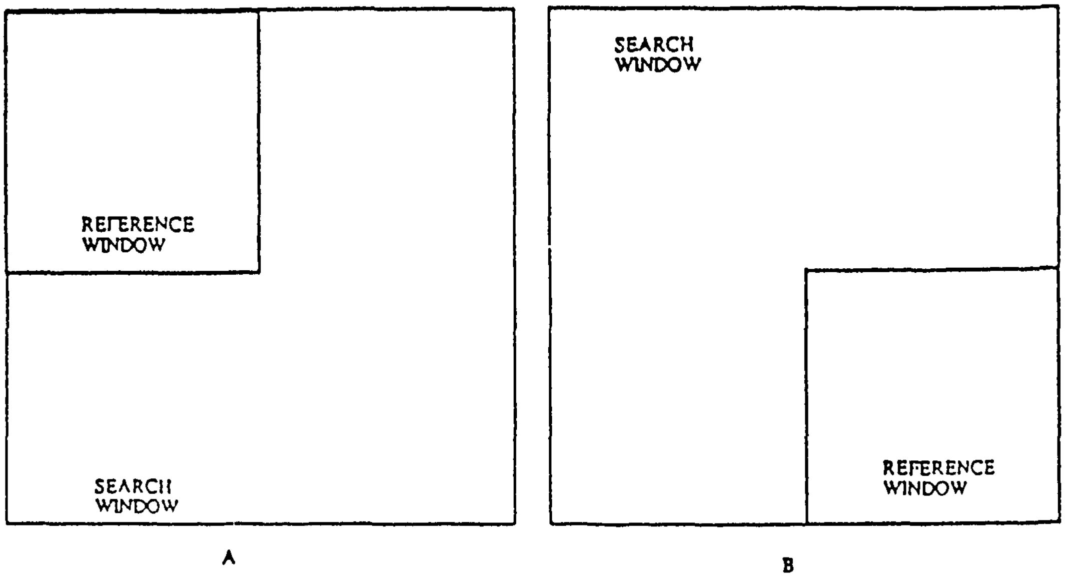


Fig. 13. First (a) and last (b) placement of reference window

8. Iteratively correlate sub-scenes for attitude angles

This part of the procedure determines the 'best' satellite attitude angles, based on new match values calculated between the reference and search windows. The value for the timing error has been found in the previous step. The WVS can now be remapped (iteratively) using the new time and varying satellite attitude angles. The attitude angles are first changed by 5 milliradians. There are three angles (the roll, pitch and yaw) which may have positive, negative or zero (with positive being counterclockwise about the principal axis, assuming a right hand system). This results in a total of 27 combinations of attitude angles. For each set, the WVS has been recalculated and compared to the satellite binary sub-scene to obtain a SAD error. The set of attitude angles with the smallest error is then changed by 0.5 milliradians. The procedure is repeated for these 27 attitude sets. The attitude angles with the smallest SAD error from this group is then changed by 0.1 milliradians. After these 27 comparisons are completed, the attitude angles which result in the smallest error between the WVS reference window and the satellite search window are chosen as the attitude angles to be used to correct the image for distortions created by the non-zero roll, pitch and yaw.

9. Applying new time and attitude angles

Having obtained the time and attitude corrections, they can now be applied to the image. This is accomplished through existing steps in Avian (Appendix A). Forward runs a forward navigation for every 16th line and pixel value and Paint creates a sub-scene of the image using the updated ephemeris information. After running Forward, the image is then mapped into one of thirteen projections using Realmap. The projection used depends on the needs of the user and the locations of interest.

V. EXPERIMENTAL DESIGN

The automatic Avian procedure was tested on ten AVHRR passes. All ten passes covered the eastern North Pacific Ocean and contained portions of the North American continent. Before navigating automatically, the passes were navigated with the original Avian technique utilizing the interactive landmarking procedure. The passes were navigated with zero, one and then four landmarks (GCPs are referred to as landmarks in this procedure). All earth locations were completed by the author, which is important as there is an element of subjectivity within the interactive landmarking procedure. It can be difficult for an inexperienced operator to choose landmarks accurately and, if chosen inaccurately, to recognize this fact and re-select the landmark. Having the same operator perform these tests may tend to reduce the variability that would be introduced by different operators. Accuracies will vary depending upon operator, as is shown by Bethke (1988). Bethke lists three major constraints influencing the selection of navigation landmarks (GCPs) and the resulting accuracy of the imagery. These deal with the ability of the operator to choose a 'good' landmark, the quality of the imagery and the resolution of the imagery. By limiting the experiment to a single operator, the variability of the first constraint is minimized.

A. METHODS OF NAVIGATION

Tapes were navigated without landmarks, that is the timing error and the attitude angles were assumed to be zero. One reason for navigating the imagery this way was to obtain approximate offsets before corrections were made to the time and attitude angles so that the size of the reference and search windows could be determined. It was necessary to guarantee that shoreline in the search window would also appear in the smaller reference window. Secondly, this gives reference earth location accuracy values to serve as controls for the automatic method.

Avian was run again on the passes using one landmark. This was done so that images navigated with one landmark could be compared to imagery navigated with one window correlation through the automatic Avian navigation procedure. One landmark is often used in other studies to correct for timing errors (Table 1).

The ten passes were then navigated using four landmarks. This approach was chosen since in previous papers the accuracy often improved when using more than one GCP. Four was the optimum number of GCPs for all of the passes. More landmarks were not used as some images had a good deal of cloud cover which limited the number of GCPs. The landmarks were chosen as carefully as possible. Twiddle displays approximate error vectors when the Twiddle procedure is initiated (Appendix A). This allows the operator to recognize poorly selected landmarks and to return to the earlier Pickem procedure and re-select the landmark. However, when choosing more than one landmark it is much harder to obtain all of the landmarks as accurately as when a single landmark has been chosen. When one landmark is selected, it can be repeatedly selected until the error vector is small. When more landmarks are be selected concurrently, it is much more difficult for the operator to choose which landmarks are are causing the greatest effect on the error vectors. Thus, it is harder to select each of the multiple landmarks as accurately as a single landmark.

The ten passes were then navigated using the automatic Avian procedure. The windows were selected in clear areas of the coast, as close to the centerline of the image as possible. However, several of the passes were either too cloudy over the center of the image to allow this or the land area was located to the right half of the image. Each navigation used one window area to obtain the corrections to the time (yielding an along track error) and to the roll, pitch and yaw.

B. ACCURACY DETERMINATION OF NAVIGATIONS

The accuracies of the individual navigations were determined by utilizing a feature contained in the Paint procedure of Avian. Once the updated time and attitude angles were obtained (either by Pickem and Twiddle in the original Avian or by the automatic Twiddle in automatic Avian) sub-scenes could be created by the Paint procedure. The latitude/longitude of particular points (known landmark locations were used) can then be obtained simply by moving a cursor to the point in question. This latitude/longitude position is determined by the mapping procedures described in Appendix B which utilize the updated ephemeris data. The difference between the navigated position and the actual position was then determined. This difference was obtained using the following equation from Bowditch (1984) for the great circle difference:

$$D = \cos^{-1}[(sinL_1 \times sinL_2) + (cosL_1 \times cosL_2 \times cosDL_0)] \qquad (3)$$

where

L_1	latitude of navigated point
L_2	true latitude of the point
DL_0	difference between longitudes

D distance (in degrees)

There are more accurate methods of determining the distance between two points; however, as the resolution of AVHRR imagery is 1.1 km, this method is more than adequate for the needs of this experiment.

The points chosen to validate the accuracy were selected as evenly distributed over the pass as possible. In cases were the entire coastline was not contaminated by clouds, points were selected over the entire range of the coastline and were evenly spaced. In images with partial obstruction due to clouds, the clear coastline areas were used, again with even distributions of the points throughout these areas. As different numbers of points were used for different navigations and different images, the standard error of the mean was determined for each navigation and then for each method (i.e., for zero, one and four landmarks and for automatic Avian) to statistically weight the determinations, as described in the data analysis section.

Locating the points on the image to validate the accuracy of the navigation is a subjective step in a sense. Locating each landmark to the exact pixel can be very difficult, particularly towards the lateral sections of the image. Distortions due to the satellite projection and the changing areal content of the pixels (Fig. 14) make this difficult at times. For this reason, points were selected as evenly distributed as possible and then averages of these points obtained to validate the earth locations.

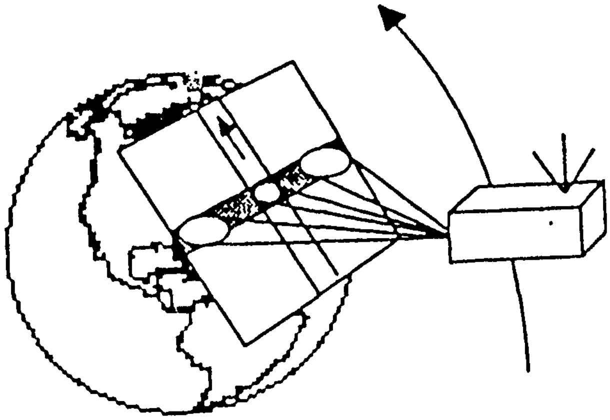


Fig. 14. Illustration of the effect of scan geometry on pixel size (Bethke, 1988)

VI. RESULTS

The results of navigating ten different AVHRR images by the original Avian procedure and the automatic Avian procedure are presented in Table 2. The columns labelled zero landmarks, one landmark and four landmarks were navigated with the original Avian procedure and the last column is the results of the automatic Avian procedure. The date of the satellite pass is given in the first column.

The average accuracy with zero landmarks was approximately 12 km, with the lowest accuracy of a pass being 10.35 km and the highest at 15.03 km. These values are not as accurate as the procedure developed by Brunel and Marsouin (Table 1) which does not use any GCPs. Their method does however, make use of time errors obtained from NOAA/NESDIS which would eliminate much of the along track error resultant from clock offset. The original Avian procedure with zero landmarks does not correct for the time offset, thus the accuracies are not as small. The third column presents the accuracies of the orignal Avian technique using one landmark. The average accuracy was approximately 2 km, with 1.05 being the minimum and 3.17 the maximum. These values compare favorably with techniques developed by Ho and Asem (1984) and Emery and Ikeda (1984) which also utilized one GCP. The previous studies resulted with accuracies ranging from 1.5 to 2.0 km. Avian with one landmark produces accuracies within this range. The original Avian procedure utilizing four landmarks is presented in the fourth column. The average accuracy improved to 1.5 km, with a minimum accuracy of 1.00 km and a maximum of 1.97 km. The last column illustrates the accuracies produced with the automatic Avian procedure. The average accuracy was approximately 1.3 km, with a minimum accuracy of 1.00 km and a maximum of 1.69 km. Automatic Avian compares very well with the method developed by Brush (1988). Brush's method utilized a 'nudge' to bring the coastline within an image into alignment with a reference coastline. This produced accuracies of 2 to 3 km. The automatic Avian procedure utilizes shoreline matching to update the ephemeris data and produces imagery navigated close to the resolution of the data.

Table 2. **ACCURACIES OF AVIAN METHODS (IN KM).** The columns labelled 0, 1 and 4 landmarks were navigated with original Avian. The last column used the automatic Avian procedure.

Tape date	0 landmarks	1 landmark	4 landmarks	auto Avian
05-15-86	12.44	2.12	1.63	1.00
05-16-86	11.45	2.67	1.97	1.69
06-17-87	14.35	1.96	1.02	1.48
06-28-87	12.65	3.17	1.00	1.64
07-06-87	12.39	1.64	1.13	1.37
07-07-87	11.57	1.05	1.78	1.43
07-14-87	12.08	1.13	1.40	1.04
07-16-87	15.03	1.52	1.45	1.23
07-18-87	10.35	1.84	1.74	1.13
04-23-88	10.91	2.13	1.47	1.22
Mean	**12.32**	**1.92**	**1.46**	**1.32**

In order to test whether the automatic Avian procedure is more accurate than the original Avian with the interactive landmarking, the data sets were statistically compared. As stated earlier, since varying numbers of points were used to determine the accuracies of different cases, the standard error of the mean was determined for each method (Table 3). If two populations have the same variance the Student's t statistic can be used to test whether the true means of two populations are the same ($\mu_1 = \mu_2$) However, if it is not known that the variances are equal, then the problem can be approximated by the Student's t, using the procedure described as the Fisher-Behrens problem (Hamilton, 1964). With this procedure the degrees of freedom for the Student's t is given in equation 4.

$$f = \frac{\left[\frac{(\sigma_1^2)}{(n_1)} + \frac{(\sigma_2^2)}{(n_2)} \right]^2}{\left[\frac{(\sigma_1^4)}{(n_1^2 \times (n_1 - 1))} + \frac{(\sigma_2^4)}{(n_2^2 \times (n_2 - 1))} \right]} \qquad (4)$$

Using this test, the automatic Avian procedure is not more accurate than the original Avian with four landmarks at 95% confidence. The automatic Avian procedure is more accurate than the original Avian procedure utilizing one landmark at 95% confidence and of course, was significantly better that the original Avian with zero landmarks.

Table 3. MEANS AND STANDARD ERROR OF THE MEANS OF IMAGE NAVIGATION

statistic	0 landmarks	1 landmark	4 landmarks	auto Avian
Mean	12.32 km	1.92 km	1.46 km	1.32 km
S.E. of Mean	0.786 km	0.340 km	0.262 km	0.246 km

Although the original Avian procedure with four landmarks produced earth locations of approximately the same accuracy as the automatic Avian procedure in this experiment, this is somewhat misleading. As previously stated, the original Avian procedure can produce accuracies that vary greatly depending upon the level of training and the expertise of the operator. Many earth location studies have been completed where the accuracy varied from 2 to 5 km. The automatic Avian procedure has eliminated the subjectivity inherent with operator interactive acquisition of landmarks. Presently, during the navigation procedure, the interactive procedure is limited to the operator selecting a clear portion of the sub-scene that contains shoreline to locate the reference and search windows. The timing error and attitude angle errors are then calculated automatically. Figures 15 and 16 shows the satellite and WVS sub-scenes overlain before and after the timing error and attitude errors have been determined. The WVS has been calculated utilizing the ephemeris data in Figure 15 without the corrections made, and in Figure 16 the timing and attitude angle errors have been corrected, illustrating that these corrections do improve the ephemeris data for this satellite pass.

The results of the earth location with automatic Avian compare favorably with recent studies (Table 1). Accuracies are approaching the resolution level of the imagery itself and are obtained with little human interaction. Results of the navigation with one landmark using original Avian provide accuracies of approximately 2 km. This confirms that the conclusions reached by Emery et al. (1989) that navigation can be completed accurately using one GCP while ignoring attitude angle errors. Figure 15 illustrates that the along-track error is the largest offset between the two shorelines. Additional landmarks used with the original Avian procedure result in slightly more accurate navigation, as a result of obtaining more accurate roll, pitch and yaw errors and applying these corrections with the navigation procedure. The automatic Avian procedure produces accurately navigated imagery while eliminating the subjective nature of the original Avian procedure. Accuracies equivalent or better than recent studies has been accomplished without relying on stored sets of GCPs, as Ho and Asem (1984) or updates of timing errors of the satellites from outside sources, as Brunel and Marsouin (1987).

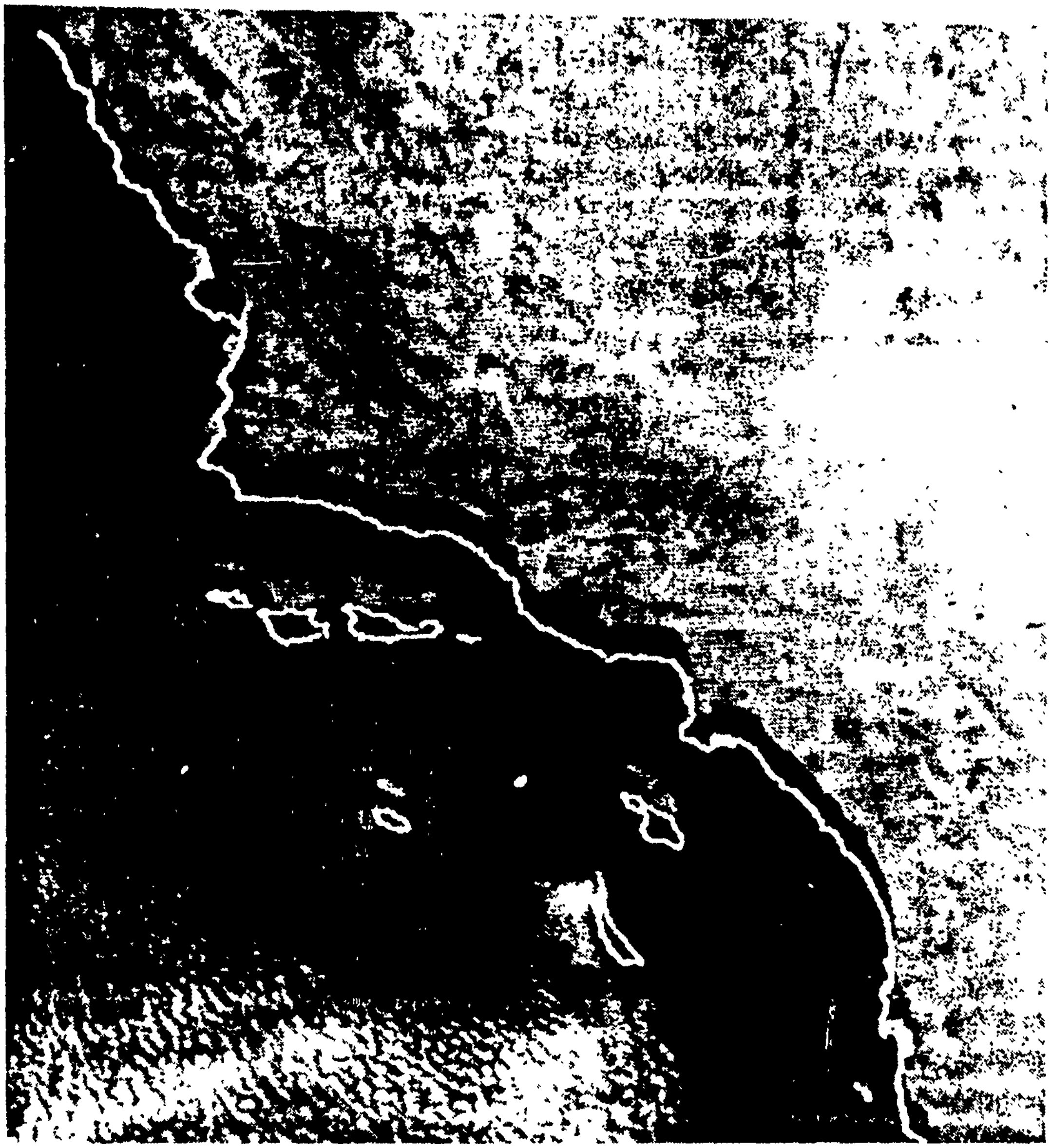


Fig. 15. WVS and satellite sub-scene overlaid before corrections: The WVS is depicted as the white shoreline.

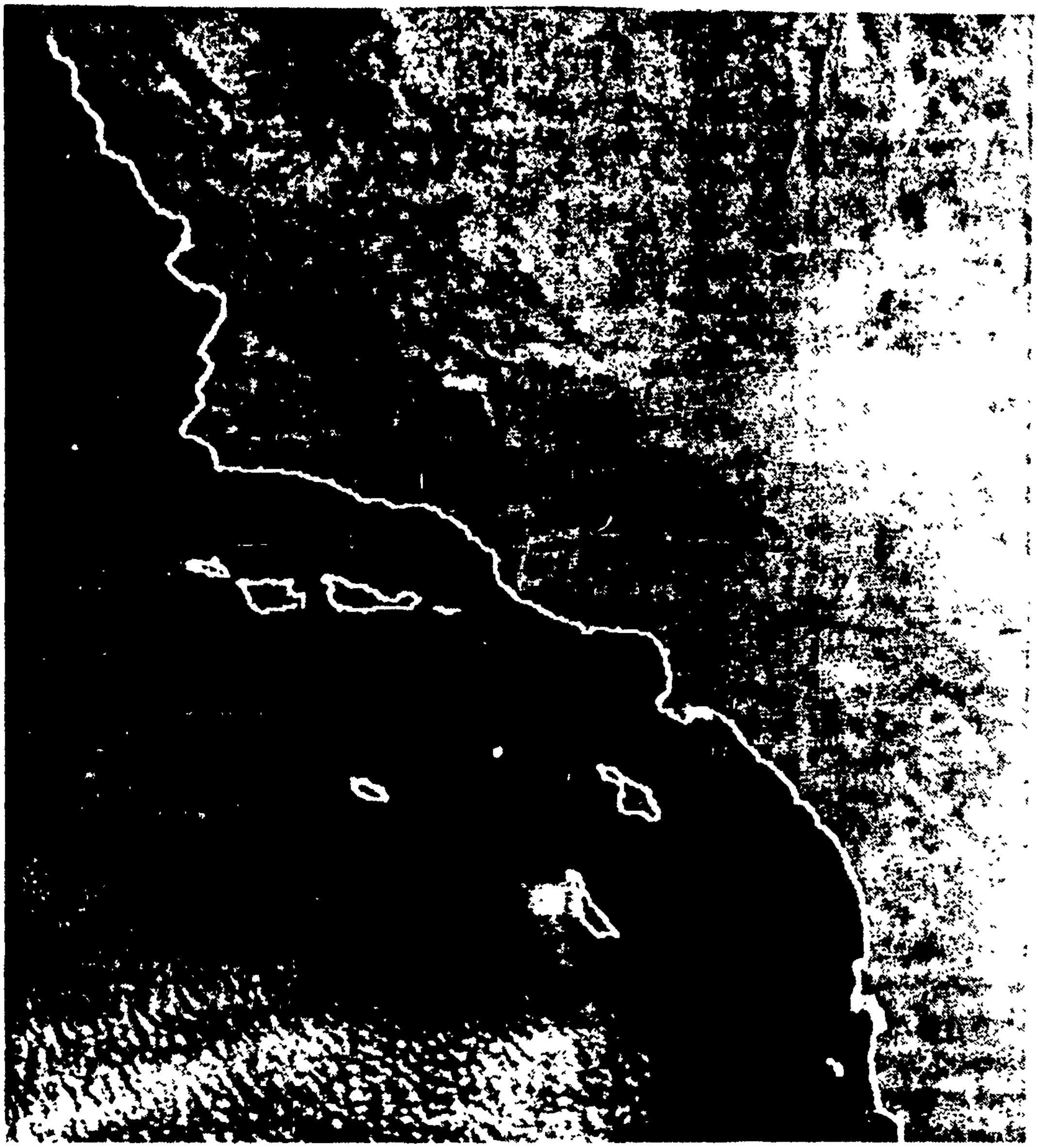


Fig. 16. WVS and satellite sub-scenes overlaid after corrections: The WVS is depicted as the white shoreline.

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

An automatic method of navigating AVHRR imagery has been developed and tested successfully in this paper. This automated procedure utilizes DMA's WVS database as a reference base, binary shoreline images and the Sum of Absolute Differences (SAD) matching technique to solve for errors in timing from the satellite's clock and in the satellite attitude angles. When these errors have been calculated the corrections are added to the ephemeris data and then the existing capabilities of Avian produce navigated imagery through a series of coordinate transformations, resulting in imagery with standard geographic coordinates. It has been shown that this automated method can produce navigated imagery at least as accurate as the original Avian procedure with four landmarks. The automatic method also eliminates the subjectivity inherent with interactive landmark acquisition, so that the navigation can be performed by inexperienced operators and still result with highly accurate results.

Another advantage of this automatic navigation method is that a set or library of GCPs do not need to be accumulated and stored to be used in the navigation process. The original Avian procedure had a set of landmarks that had to be accumulated over a period of time. The automatic procedure developed by Ho and Asem (1984) had to rely on a set of stored landmarks, based on gray level gradients. The automatic Avian procedure matches the shoreline from the satellite image with the shoreline presented in the WVS database. Because WVS covers the entire world, there is no problem in obtaining shoreline data.

Currently, the automatic Avian procedure takes approximately the same amount of time (approximately one hour) to complete as the original interactive procedure although the time spent interactively selecting landmarks can vary greatly. The image navigation in this study were performed on a DEC Microvax II and time constraints will vary depending on computer system used. Selecting four or five landmarks accurately can be very difficult, and an operator may have to re-select landmarks several times to be able to produce accuracies of navigated imagery similar to that produced in this paper. There is also the benefit of less interactive steps with the automatic procedure. Once the operator has selected the window area, the automatic Avian technique has no other interactive steps. This allows the operator to complete other work while the corrections to the time and the attitude angles are calculated. The original Avian technique involved the operator interactively throughout the procedure. At the present time, the automatic procedure has not significantly reduced the amount of time spent navigating imagery, but it has reduced the amount of time that an operator is interactively involved in the process.

B. RECOMMENDATIONS

There are several procedural steps that can be improved to produce more consistent and timely earth location.

1. Placement of the matching windows

A comprehensive study to determine whether or not the placement of the reference and search windows within the satellite image has an effect on the accuracy of the navigation would extend this research. It is likely that placement of the window along the lateral extremes of the image where there is more distortion will produce reduced accuracies. Individual passes would need to be navigated several times with the windows located at varying locations along the coastline to explore the impact of window location on accuracy.

2. Development of multi-window techniques

The original Avian procedure twiddles up to 16 GCPs (or landmarks) at one time to obtain the optimum attitude angle corrections. Development of an automated procedure utilizing more that one set of reference and search windows may improve the accuracy of the navigation over the entire pass. If the study of window placement determines that lateral placement of the window increases accuracy, a multi-window technique would allow placement of windows at varying distances from the centerline, possibly improving accuracies.

3. Fully automate the Avian procedure

Development of a fully automatic navigation procedure that would produce accurate earth location would be very beneficial. Reduction in the amount of time that the procedure needs operator attention is desirable. A possible approach would be to map the WVS data into the POS projection using the ephemeris data provided on the tape. Through this work, it is known that the satellite shoreline would be within approximately 12 km of the shoreline from the WVS. Depending upon the number of window areas desirable and the number of points in the WVS for that image, an arbitrary number can be selected that would pick a point out of the WVS. This point would be used as the midpoint for the reference and search windows. A sub-scene would first need to be created at full resolution which contained these areas. It could be thresholded and edge enhanced automatically. The SAD matching values would then be calculated between the reference and search windows. A critical value would be set (this would need to be determined ahead of time through experimentation) and if the error obtained from the SAD procedure was below this critical value, this window area could be used. If the window area is rejected, the procedure would skip down the recommended number of points within the WVS data and repeat the procedure. This could be repeated until a satisfactory window area is found or if a multi-window technique is being used, until the number of windows specified is found. A fully automatic Avian procedure could produce accurately navigated imagery with a minimal amount of operator time involved. Thus, earth located satellite imagery would be readily available for scientific use. As the use of digital polar orbiter imagery increases, automated navigation techniques, like the method described here, will become even more critical for the effective use of NOAA and DMSP data.

4. Compare with new navigation technique

The Department of Oceanography at the Naval Postgraduate School has recently acquired a satellite image navigation software package from the University of Miami. This navigation system was not online at the time this thesis was printed. This navigation system also uses coastal boundaries to navigate satellite images. A detailed comparison between the automatic Avian procedure and this new package needs to be performed in order to determine the uses of these procedures at the Naval Postgraduate School. Accuracy of navigation (including the accuracy of the shoreline reference base used), amount of time spent performing navigation and user friendliness of the two systems need to be compared.

APPENDIX A. ORIGINAL AVIAN PROCEDURE

Avian is a software package that has been implemented at NPS for navigation, calibration and image analysis of polar orbiter AVHRR data. Avian is presented here in the same sequence as to the user at the terminal, a step by step procedure that follows a menu given in the IDEA (Interactive Digital Environmental Analysis) lab. The theoretical background of the navigation process is reviewed in Appendix B. There are eight steps, some of which may or may not be needed during a particular navigation.

1. Glean

This is the first step and is required for all tapes that are to be navigated. Gleaning retrieves all of the satellite image data from a magnetic tape. Glean produces three files, an image information file (INF), a raw calibration data file (CAL) and an image data file (SCN). After Glean is run , a routine called Calib is automatically run that produces a file containing actual calibration coefficients (i.e., temperature coefficients for channels 4 and 5 in Kelvin, albedo coefficients for channels 1 and 2 and radiance coefficients for channel 3).

2. Overview

This routine allows the user to view the entire image on a display monitor, at any channel (although at low resolution). This step is required if an image is to be navigated as sub-images at full resolution are created from this in the Pickem and Paint steps of Avian. An overview (OVR) image file is created.

3. Pickem

This routine is used to supply a set of points which will be used to create a mapping from the image coordinate system to the geographic coordinate system. The INF, SCN and OVR files are required to run Pickem. The overview image is presented and the operator then chooses small areas at which known landmarks exist (points, headlands and sharp edges along the coast). These areas are expanded by a factor of four and the operator then picks the landmark location by cursor. When selecting these landmarks the operator must choose landmarks that have their positions stored in a library of GCPs which is stored within the program. A list of these is available for operator reference. The landmarks are placed in a landmark (LND) file, to be used in the routine following Pickem.

4. Twiddle

Twiddle is an iterative procedure which determines four unknown quantities, the timing error (which accounts for an along-track error) and the satellite attitude angles (the roll, pitch and yaw). The Twiddle procedure is shown in Figure 17. This step uses the INF and LND files that were produced in the earlier steps. Twiddle uses positions of landmarks (GCP's) which were manually selected in the Pickem procedure. The pixel and line number (or row and column) were saved for each selected landmark. Twiddle has a library of GCP's in which the latitude and longitude of each GCP is stored. For each GCP chosen, the latitude and longitude of the corresponding stored 'correct' positron is converted to line and pixel numbers. This conversion utilizes the ephemeris data and results is a 'navigated' GCP position, which the picked landmark will be compared. The conversion from geodetic coordinates to satellite image coordinates is a reverse navigation and is detailed in section E of Appendix B. The difference between the line number of each landmark and its corresponding navigated GCP position is then obtained. The average of all of these offsets is then calculated. This average line offset yields the timing offset as the radiometer on the NOAA satellites scans six lines per second. The offset in time is then added to the time given in the ephemeris data to update and time and thus correct the along-track offset.

To obtain the corrections for the satellite attitude angles several steps are taken. The first step is to calculate the error for each landmark (i.e., the difference in the position of the landmark GCP and the navigated GCP). These errors are calculated and an error vector is saved and also displayed on the monitor for each GCP. At this point, any landmark which appears to have a large error can be erased (using the procedure described under Utilities) and re-selected in the Pickem procedure. These error vectors are first obtained assuming zero errors in the roll, pitch and yaw. The landmarks are then 'twiddled', that is each of the three attitude angles are changed by fixed amounts in both directions. There are a total of 27 combinations of differing roll, pitch and yaw that are calculated (including the position of zero error in the roll, pitch and yaw). The 'best' roll, pitch and yaw combination is chosen by selecting the attitude angles which produce the minimum offset between the landmarks and navigated GCP's. These angles and the time correction are saved in an updated INF file which will be used to navigate the image in the Forward process.

5. Forward

This routine produces a forward navigation of the image, as detailed in Appendix B. It utilizes the latest INF file and creates a NAV (navigation) file. The navigation

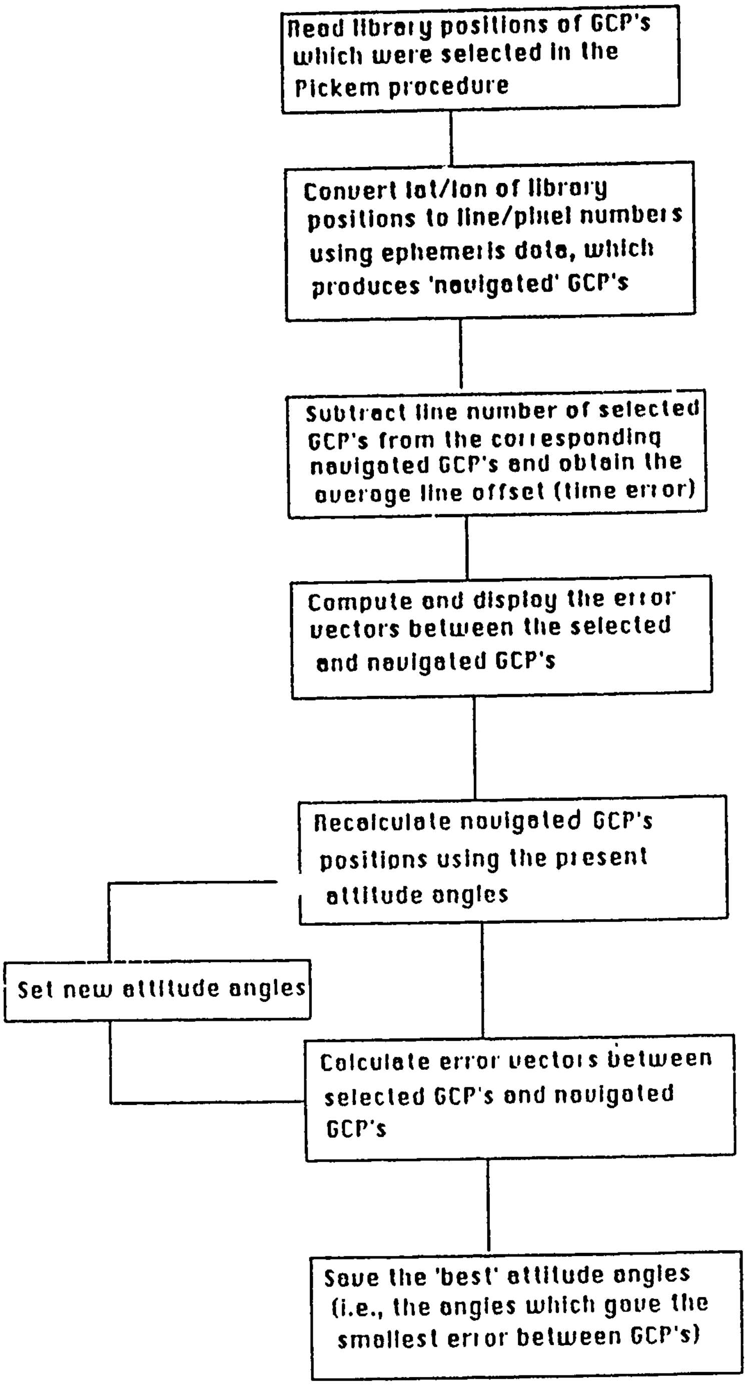

Read library positions of GCP's which were selected in the Pickem procedure
Convert lat/lon of library positions to line/pixel numbers using ephemeris data, which produces 'navigated' GCP's
Subtract line number of selected GCP's from the corresponding navigated GCP's and obtain the average line offset (time error)
Compute and display the error vectors between the selected and navigated GCP's
Recalculate navigated GCP's positions using the present attitude angles
Set new attitude angles
Calculate error vectors between selected GCP's and navigated GCP's
Save the 'best' attitude angles (i.e., the angles which gave the smallest error between GCP's)

Fig. 17. Twiddle procedure

file created contains the navigational values for every 16th line and pixel for the entire image. This file is created so that a latitude/longitude grid may be overlaid on a satellite image. Also this file provides a computationally quick method to obtain latitude/longitude values at different pixel locations when lower accuracy will suffice.

6. Paint

The Paint routine has two purposes. One is to validate the satellite image navigation and the other is to display sub-scenes of the satellite image. To aid in validation of the navigation, Paint performs a number of useful functions on full resolution sub-scenes of the image. Coastlines and political boundaries may be overlain on a sub-scene as well as latitude-longitude grids. The navigation values of any point can be displayed, allowing the operator to check how accurately the image has been navigated. Paint uses the INF, SCN, NAV and CAL files.

7. Realmap

This routine allows the user to map the satellite image to one of 13 different projections. These projections are:

Mercator
North Polar Stereographic
South Polar Stereographic
Northern Hemisphere Lambert Conic Conformal
Southern Hemisphere Lambert Conic Conformal
Cylindrical Equidistant
Azmuthal Equidistant
Modified Cylindrical Equidistant
Universal Transverse Mercator
Transverse Mercator
North Orthographic
South Orthographic
Gnomic

This enables the user to see the image in a convenient projection and utilize the data easily and efficiently.

8. Utilities

This routine enables the user to examine and manipulate the data or information files created throughout the Avian process. Landmarks can be deleted from the

LND file if they are not located accurately enough. Information, such as the updated ascending node position and satellite attitude angles can be checked through this procedure as well.

APPENDIX B. NPS SATELLITE IMAGE NAVIGATION

A. COORDINATE CONVERSIONS

Navigating satellite images is equivalent to developing a mapping function from satellite image coordinates into geographic coordinates. This mapping is a complicated function and while it is possible to undertake in a single step, it is easier to understand with steps using intermediate coordinate systems. This outline of satellite image navigation is taken from Burks (1988).

1. Coordinate systems

Developing the mapping from satellite image coordinates to geographic coordinates (and also the reverse process) involves the following six coordinate systems:

1. satellite image coordinate system
2. scanner coordinate system
3. satellite orbit plane coordinate system
4. geocentric orbit plane coordinate system
5. geocentric coordinate system
6. geodetic coordinate system

For brevity the 'satellite orbit plane coordinate system' will be referred to as the 'satellite coordinate system' and the 'geocentric orbit plane coordinate system' will be referred to as the 'orbit plane coordinate system'. The first and the sixth coordinate systems in the list are two-dimensional coordinate systems. The three-dimensional coordinate will be in cartesian form, allowing easy matrix manipulations. Occasionally, spherical coordinates will be referenced and latitude, longitude and radius coordinates will be used.

In the cartesian system, the equatorial (zero latitude) plane coincides with the cartesian x-y plane with the x-axis pointing through zero longitude, the y-axis at 90 degrees from the x-axis, in the clockwise direction. The z-axis is normal to the equatorial plane, pointing in the direction of increasing longitude in a dextral sense, thereby defining a right-hand coordinate system.

a. *The satellite image coordinate system*

This coordinate system is defined by the image scanning instrument. The origin (1,1) is the first pixel observed on the first line of the image. The first line of an image is arbitrary, however in Avian the first line is chosen to coincide with the earliest time (fig. 18). Note that for a descending orbit, the first line will be shown as the northernmost point on the image; while for an ascending orbit, the first line will be shown as the southernmost point on the image. The x-coordinate is the pixel number, increasing along a single scan line. The y-coordinate is the scan line number, increasing along the image. As mentioned earlier, this system is a two-dimensional system with no spherical (or polar) coordinate counterpart.

b. The scanner coordinate system

This coordinate system is generally aligned with the satellite coordinate system, described later in this section. Its origin sits in the scanning instrument on its rotational axis. The x-axis is normal to the scanners' rotational axis pointing in the direction of the spacecraft motion. The z-axis points straight downward in the general direction of the center of the Earth. The y-axis completes a dextral (right handed) coordinate system.

c. The satellite coordinate system

This coordinate system has its origin at the instruments scan axis. The x-axis points in the coordinate of the motion of the satellite. The z-axis points through the Earth's center and the y-axis completes a dextral coordinate system. The x-z plane coincides with the instantaneous satellite orbit plane (fig 19). It differs from the scanner coordinate system by a rotation through the satellite attitude angles (roll, pitch and yaw).

d. The orbit plane coordinate system

This coordinate system is geocentric, with the x-axis pointing through the perigee of the orbit. The z-axis is normal to this plane and points in the direction of the satellite's angular velocity. The y-axis completes a dextral coordinate system. In its spherical coordinate analogue, the longitude is called the true anomaly. The center of the Earth is one of the assumed elliptical orbit foci (fig. 20).

e. The geocentric coordinate system

This coordinate system is based on the Earth. The z-axis is in the direction of the Earth's rotational angular velocity. The x-y plane is normal to the z-axis, anchored at the Earth's center. The x-axis points through zero longitude and the y-axis completes a dextral coordinate system (fig. 21).

f. The geodetic coordinate system

This coordinate system is the standard latitude, longitude system. It is based on the Earths' idealized oblate spherical surface. The third dimension (height above the Earth's surface) is not used for satellite image navigation (fig. 22).

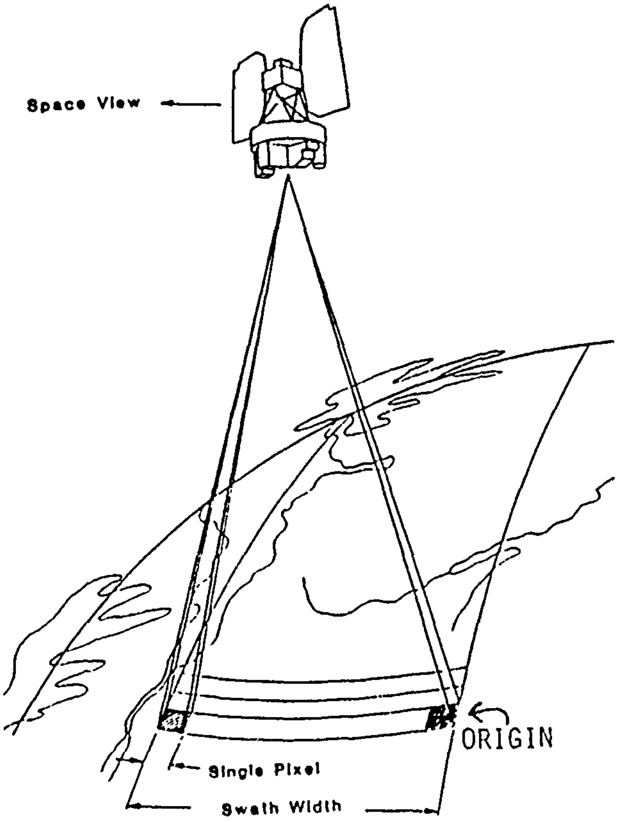


Fig. 18. Chosen origin of AVHRR imagery in Avian (Rao et al., 1990)

2. Geodetic to geocentric coordinates

Geodetic latitude and longitude are the end product of the full satellite image navigation and typical of normal map presentations. However, it is connected with the Earth's surface. This conversion changes the origin of the coordinate system from the Earth's surface to the Earth's center and changes the reference suiface from an oblate spheroid to a sphere. This converts the geodetic latitude (ϕ') and longitude (λ') to the geocentric latitude (ϕ) and longitude (λ) as shown in equation (B-1).

$$\phi = atan(\tan \phi' \times (1 - f^2))$$

$$\lambda = \lambda' \qquad (B-1).$$

Where f is the flattening of the Earth's surface (approximately 1/297). The Earth's radius (r) is a function of ϕ':

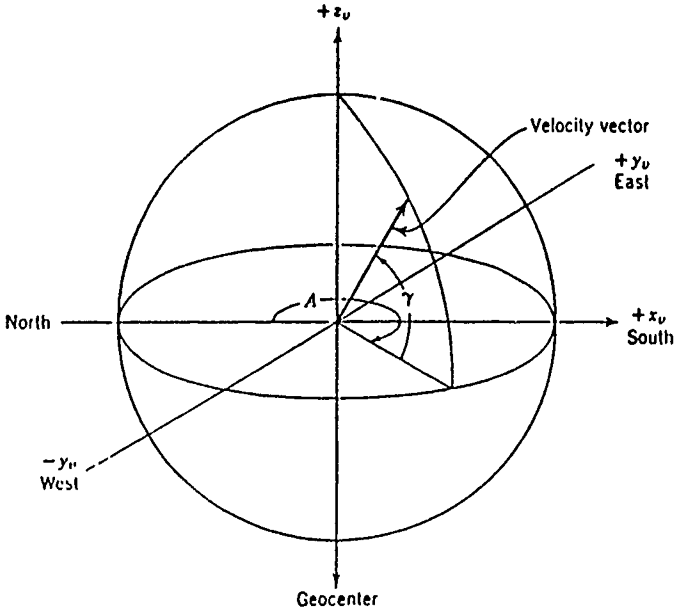


Fig. 19. Satellite coordinate system (Escobal, 1965)

$$r^2 = \frac{1.0}{\left[\frac{(\cos^2\phi)}{a^2} + \frac{(\sin^2\phi)}{b^2} \right]} \qquad (B-2).$$

where a is the semi-major axis and b is the semi-minor axis of the Earth's surface. Having the latitude, longitude and radius, a cartesian geocentric position vector for a point on the Earth's surface can be constructed.

3. Geocentric to orbit plane coordinates

This conversion is a well-defined three-dimensional rotation using the orbital elements at a given time. The rotation matrix G is given as:

$$G = \begin{bmatrix} (cp \times cn - sp \times sn \times ci) & (cp \times sn + sp \times cn \times ci) & (sp \times si) \\ (-sp \times cn - cp \times sn \times ci) & (-sp \times sn + cp \times cn \times ci) & (cp \times si) \\ (sn \times si) & (-cn \times si) & ci \end{bmatrix} \qquad (B-3)$$

where:

cp	cosine of the argument of perigee
sp	sine of the argument of perigee

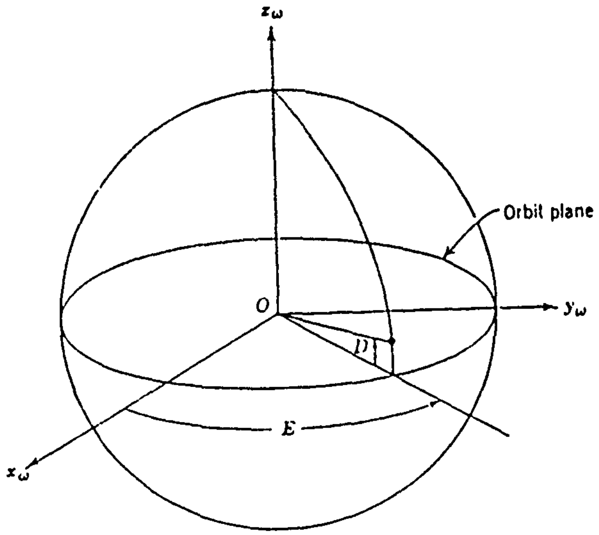


Fig. 20. Orbit plane coordinate system (Escobal, 1965)

cn	cosine of the geocentric ascending node longitude
sn	sine of the geocentric ascending node longitude
ci	cosine of the orbit inclination
si	sine of the orbit inclination

This rotation is a combination of rotating through each angle individually, in the correct order.

The reverse transformation multiplies an orbit plane coordinate position vector by the inverse of the matrix G. Since the rotation matrix is orthogonal, this is equal to multiplying by the transpose of the matrix.

4. Orbit plane to satellite coordinates

This transformation is one of the more difficult used, as it involves both a rotation and a translation. Figure 23 depicts a satellite to geocentric transformation (Saufley, 1982). In this figure, c and d are angles which, if known, determine the position of a pixel on the ground. Both of these coordinate systems use two axes to define the orbit plane, differing as to which two they use. The position of the reference longitude in the orbit plane also differs by a rotation in the orbit plane through the argument of the perigee.

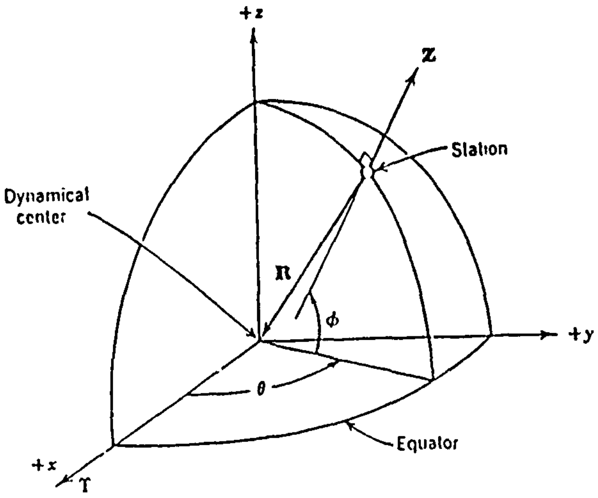


Fig. 21. Geocentric coordinate system (Escobal, 1965)

Thus the full rotation is a multiplication of these two matrices (eqn. B-4).

$$R(x') = XV(x) \qquad (B-4)$$

where

$$X = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \qquad (B-5)$$

$$V = \begin{bmatrix} cw & sw & 0 \\ -sw & cw & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (B-6)$$

thus,

$$R = \begin{bmatrix} 0 & 0 & -1 \\ cw & sw & 0 \\ sw & -cw & 0 \end{bmatrix} \qquad (B-7)$$

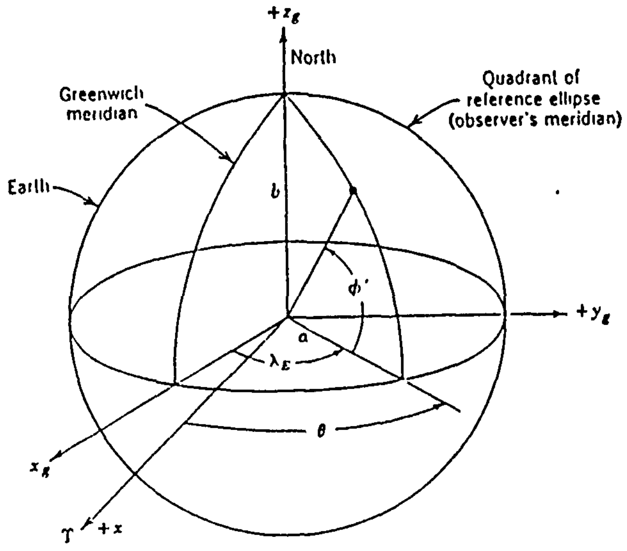


Fig. 22. Geodetic coordinate system (Escobal, 1965)

where

cw cosine of the argument of perigee

sw sine of the argument of perigee

To translate the centers of the coordinate systems, the plane triangle defined by the center of the Earth, the satellite and the observed point is used. Three vectors from the sides of this triangle are:

S the satellite vector, which is the geocentric position of the satellite

P the pixel vector, which is the geocentric position vector of the observed point

V the view vector, which is the vector from the satellite to the observed point

From these relationships you obtain:

$$S + P = V \qquad (B-8)$$

The pixel vector (P) and the satellite position vector (S) are known, so once the two vectors are rotated into the same coordinate system the third can be found (Fig. 24).

The reverse transformation multiplies a satellite coordinate position vector by the inverse of the matrix (R). Since the rotation matrix is orthogonal, this is equivalent

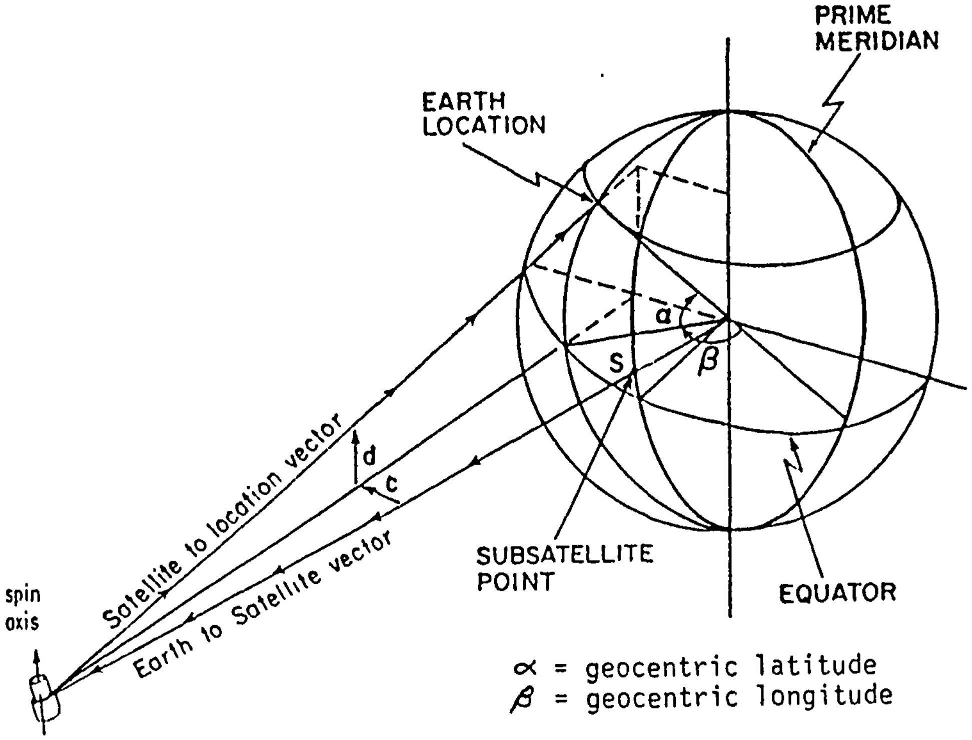


Fig. 23. Satellite to geocentric transformation (Saufley, 1982)

to multiplying by the transpose of the matrix R. The vector addition also still holds, except that the satellite and pixel vectors are known, giving:

$$P = V - S \qquad (B-9)$$

5. **Satellite to scanner coordinates**

The scanner is solidly anchored onto the satellite, but the satellite may not be lined up in orbit. The small rotations from a true attitude are the satellite attitude angles (roll, pitch and yaw) which represent rotations around the x-, y- and z-axes respectively. Converting from satellite to scanner coordinates thus involves rotating the position vector through the roll angle, the pitch angle and then the yaw angle. Each rotation is represented by a rotation matrix R, P and Y respectively. The total rotation is the multiplication of these matrices:

$$A(x') = R \times P \times Y(x) \qquad (B-10)$$

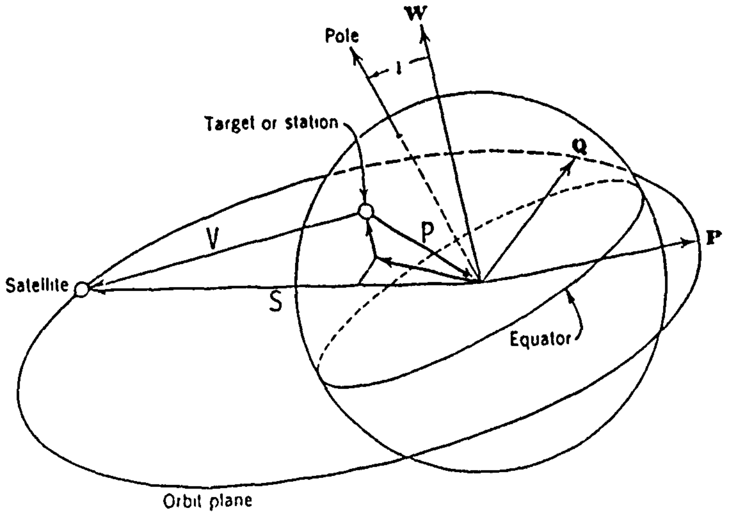


Fig. 24. Plane triangle defined by Earth center, satellite and pixel (Escobal, 1965)

where

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cr & sr \\ 0 & -sr & cr \end{bmatrix} \qquad (B-11)$$

$$P = \begin{bmatrix} cp & 0 & p \\ 0 & 1 & 0 \\ sp & 0 & cp \end{bmatrix} \qquad (B-12)$$

$$Y = \begin{bmatrix} cy & sy & 0 \\ -sy & cy & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (B-13)$$

Thus,

$$A = \begin{bmatrix} (cp \times cy) & (sr \times sp \times cy + cr \times sy) & (-cr \times sp \times cy + sr \times sy) \\ (-cp \times sy) & (-sr \times sp \times sy + cr \times cy) & (cr \times sp \times sy + sr \times cy) \\ sp & (-sr \times cp) & (cr \times cp) \end{bmatrix} \qquad (B-14)$$

where

cr	cosine of the roll angle
sr	sine of the roll angle
cp	cosine of the pitch angle
sp	sine of the pitch angle
cy	cosine of the yaw angle
sy	sine of the yaw angle

The reverse transformation multiplies a scanner coordinate position vector by the inverse of the matrix A. Since the rotation matrix is orthogonal, the inverse is equal to the transpose of the matrix A.

6. Scanner coordinates to image coordinates

This coordinate transformation is defined by the sensor model. The instrument scans the image and each pixel and line is mapped into a unit vector in the scanner coordinate system. If the scanner moves only in one plane the line number is a function of time. This is completely instrument dependent.

The AVHRR radiometer scans within a single plane (Fig. 17), where the scan angle ($scan\angle$) is a linear function of the pixel number:

$$scan\angle = 9.447117e^{04} \times pixel\,no - 0.967856654 \qquad (B-15)$$

where the zero angle points straight downward. Thus a unit vector in the scanner coordinate system can be constructed by:

$$x = 0.0$$

$$y = \cos(scan\angle)$$

$$z = \sin(scan\angle) \qquad (B-16)$$

These are the scanner coordinates of the pixel and the navigation proceeds from there.

The scanner coordinates of the pixel are independent of the image line number. Since the scanner moves from line to line purely with the satellite motion, the line number is a strict function of time:

$$t = t_0 + \frac{1}{6} \times (line\,no - 1) \qquad (B-17)$$

where t_0 is the time for the first line in the image and the units are in seconds. Note that the AVHRR radiometer scans six lines per second. When navigating in a given direction, the line number yields the time at which to calculate the orbital elements.

B. ORBITAL ELEMENT MODEL

An orbital element model predicts the orbital elements of a satellite at a certain time, given the orbital elements at an epoch time, usually once a day. The accuracy of this model is paramount in satellite image navigation, especially for polar-orbiting satellites which pass relatively close to the Earth at high rates of speed (making two revolutions per day), as opposed to geosynchronous satellites which orbit over one area or spot.

The parameters used as orbital elements may vary according to the problem under investigation and the programmers discretion. A set always contains six parameters and any set of six can be converted to any other set of six parameters. An example of orbital elements is given in Figure 25 (Saufley, 1982). The six parameters used in Avian are:

e the eccentricity is a measure of the difference of the satellites orbit from a circle

i the inclination is the angle between the satellite orbital plane and the Earth's equatorial plane

ω the argument of perigee is the angular measure of position of the orbit perigee (the point closest to the center of the Earth) along the orbit relative to the Earth's equatorial plane

v the true anomaly is the angular measure of the satellite's position relative to the orbit's perigee along the orbit

a the semi-major axis length is half the distance along the long axis of the elliptical orbit

Ω the longitude of the ascending node is the geocentric east longitude of the point where the orbit crosses the equator from south to north

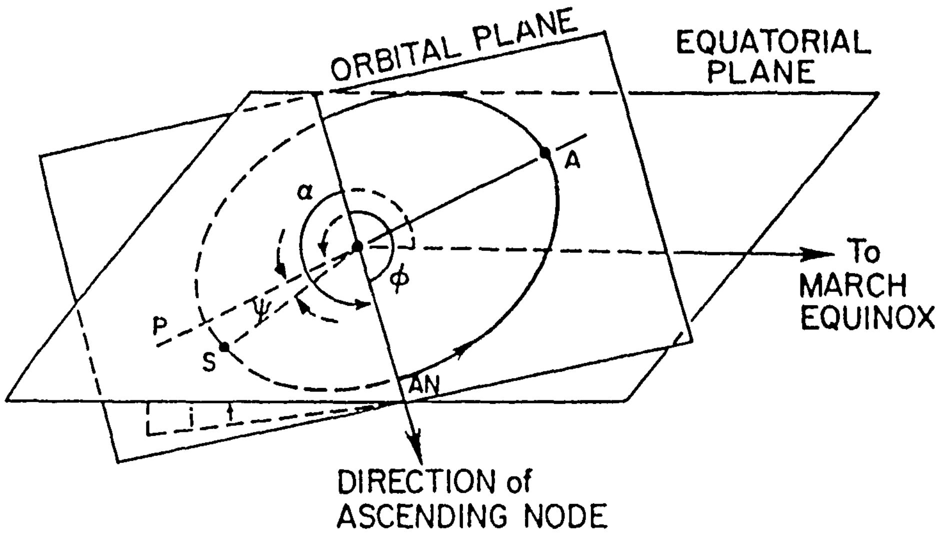
The angles measured are from the center of the Earth which sits at one of the foci of the elliptical orbit. The semi-major axis is measured from the center of the ellipse, not the center of the Earth.

The orbital elements are normally given at an epoch time which is not the time that is needed for the navigation, so the time will need to be updated. For Avian, the satellite ephemeris data is received with epoch times at 0000Z each day. These Naval Space Surveillance Center elements include:

e_0 eccentricity

i_0 inclination angle

T	Epoch (date and time)
a	Semimajor Axis
ϵ	Eccentricity
i	Orbital Inclination
ϕ	Argument of Perigee
ψ	Mean Anomaly
α	Right Ascension of Ascending Nodp



A- Apogee
P - Perigee
S - Satellite Position
AN - Ascending Node

Fig. 25. Orbital elements (Saufley, 1982)

ω_0 argument of perigee

M_0 mean anomaly, which is the angular position of the satellite relative to the orbit's perigee, along a circle circumscribing the orbit

n_0 anomalistic mean motion (rad/herg) is the angular velocity of the satellite along the circumscribing circle

n' orbital decay rate (rad/her/herg) is not an orbital element, it measures the acceleration in the mean anomaly due to non-inertial and non-perturbation effects

Ω_0 geocentric longitude of the ascending node

Projecting these orbital elements forward in time is not difficult Four do not change with time:

$$e = e_0$$

$$i = i_0$$

$$\omega = \omega_0$$

$$a = n_0^{(\frac{-2}{3})} \qquad (B-18)$$

where the latter quantity is derived from Kepler's laws of orbital motion. The longitude of the ascending node only changes due to the Earth's rotation:

$$\Omega = \Omega_0 - \Omega_e dt \qquad (B-19)$$

where Ω_e is the Earth's rotation rate. The mean anomaly of the satellite is:

$$M = M_0 + n_0 dt \qquad (B-20)$$

where dt = T - [illegible] Converting this to the true anomaly requires using elliptical geometric steps. First, find the eccentric anomaly (E) by solving Kepler's equation:

$$M = E - e \times sinE \qquad (B-21)$$

The solution is a zero order Bessel's function of the first kind. Following Smith (1980), expand the solution in terms of eccentricity and truncate it at an approximate term. For this case, keep up to the third order. The result is:

$$E = M + \sin(M \times e) + \sin(M)\cos(M \times e^2) +$$

$$(sinM - (\frac{3}{2}) \times \sin(3 \times M) \times e^3) + 0(e^4) \qquad (B-22)$$

since e is quite small, the error is negligible. The true anomaly is found by:

$$\sin v = \frac{(\sqrt{(1 - e^2)} \times \sin E)}{(1 - e \times cosE)}$$

$$\cos v = \frac{[\ \cos(E - e)]}{(1 - e \times cosE)} \qquad (B - 23)$$

Thus the six orbital elements have been obtained.

There are other problems to be resolved. The Earth's gravitational potential is not spherical so the gravitational field exerts a force on the satellite. Since the Earth's shape is well known, these perturbation effects can be calculated. However, the results show that each orbital element depends on every other orbital element, thus creating a non-linear problem. Fortunately, if integrated over one orbit, the eccentricity, inclination angle and semi-major axis do not change, which makes the problem less complex. The orbital elements with the perturbative effects (to the first order) included are given by:

$$e = e_0$$

$$i = i_0$$

$$n = \frac{(n_0)}{\left[1 + \frac{\left[(\frac{3}{2}) \times J_2 \times (1 - e^2)^{(\frac{1}{2})} \times (1 - (\frac{3}{2}) \times sin2i) \right]}{\left[a^2 \times (1 -- e^2)^2 \right]} \right]}$$

$$a = n^{(\frac{-2}{3})}$$

$$M = M_0 + ndt$$

$$\omega = \omega_0 + \frac{\left[(\frac{3}{2}) \times J_2 \times (2 - (\frac{5}{2}) \times sin2i) \right]}{\left[a^2 \times (1 - e^2)^2 \right]} dn$$

$$\Omega = \Omega_0 - \frac{(\frac{3}{2} \times J_2 \times cosi)}{\left[a^2 \times (1 - e^2)^2 \right]} \times dn - \Omega_e dt \qquad (B - 24)$$

where J_2 is the first harmonic of the Earth's gravitational potential. Here, n is the mean motion constant, not the projected anomalistic mean motion. In the unperturbed case, these two values are equal, eliminating any problems. In this case however, more care must be taken. The semi-major axis is derived from the mean motion constant, not the anomalistic mean motion. Deriving the true anomaly from the mean anomaly as described earlier, completes the orbital elements set.

Another complication is resultant from other non-perturbative forces on the satellite. These forces include atmospheric drag, gravitational influence of other bodies and radiation pressure. All of the effects can be combined into the orbital decay rate given in the epoch orbital elements which results in a correction derived from observations. This factor only changes two equations, the mean anomaly and the semi-major axis:

$$a = n^{(\frac{-2}{3})} - \frac{(4 \times a \times n')}{3} \times n \times dt$$

$$M = M_0 + ndt + n'dt^2 \qquad (B-25)$$

Unfortunately, this introduces a non-linearity between the calculation of the mean motion constant and the semi-major axis. The two quantities can be derived iteratively. First, calculate the semi-major axis using the equation:

$$a = n^{(\frac{-2}{3})} \qquad (B-26)$$

Use this value to calculate the mean motion constant of equation (B-25). This can then be used to calculate the full semi-major axis length according to equation (B-25). Iterate over these two equations until the result converges to satisfactory limits. Twice is sufficient in this case. The rest of the orbital elements follow straightforwardly the equations of the perturbative case.

C. REVERSE NAVIGATION

The reverse navigation of an image converts the latitude and longitude of an observed point to its image line and pixel number. The process steps through the coordinate systems from the geodetic to the satellite image coordinate systems (in reverse order of the list in section A of this appendix). The reverse transformations are given, but there are several problems. The time at which the pixel was observed is unknown and the orbital elements at that time are unknown also. Fortunately, the orbital elements are very nearly constant with time or vary nearly linearly with time, so the solutions can be found iteratively. That is, assume a time, calculate the orbital elements, then update the time and run through the cycle again.

Examining the problem more closely, the unknown time means that the satellite position is unknown, which affects the conversion from plane coordinates to satellite coordinates. The rotation part is fine using assumed orbital elements, but one side of the triangle is unknown. The three sides are the view vector (V), the satellite vector (S) and the pixel vector (P). The three angles are the nadir angle (n), the zenith angle (z) and the scan angle (s). Of these six quantities, only the pixel vector is known.

For a first approximation, assume that the true anomaly for the satellite is the same as the true anomaly for the pixel. This is a zero satellite attitude assumption, which is normally a good assumption as the satellite attitude normally varies only small amounts. Using this true anomaly (v sat), the satellite's orbit plane spherical coordinates are:

$$longitude = vsat$$

$$latitude = 0$$

$$radius = a \times \frac{(1 - e^2)}{(1 + e \times \cos(vsat))} \qquad (B-27)$$

These can then be converted into cartesian coordinates, the satellite vector obtained, and from equation (B-8), the view vector is calculated.

Continuing the process eventually yields a line and pixel number for the point, however to iterate successfully, the time needs to be updated after each iteration. The satellite true anomaly (v sat) varies smoothly and monotonically with time, providing the needed informatic.:. Reversing the orbital element model yields:

$$E = atan\left[\frac{(\sqrt{(1 - e^2)} \times \sin v)}{(\cos v + e)}\right]$$

$$M = E - esinE \qquad (B-28)$$

This mean anomaly is uncertain by a factor of 2π, that is, the number of satellite orbits since the epoch. Use the current estimate of the mean anomaly to get the integral multiple of 2π required and add it to the calculated mean anomaly. The time since epoch is derived by solving the quadratic equation:

$$M = M_0 + ndt + n'dt^2 \qquad (B-29)$$

Computationally, the higher order term is about eight orders of magnitude smaller than the other two terms, so the usual solution to the quadratic equation:

$$x = \frac{\left[-b + \sqrt{(b^2 - 4 \times a \times c)} \right]}{2 \times a} \qquad (B-30)$$

subtracts two similarly large quantities, losing much precision. This solution can be re-written as:

$$x = \frac{\left[-b + b \times \sqrt{(1 - 4 \times a \times \frac{c}{b^2})} \right]}{(2 \times a)} \qquad (B-31)$$

The second term in the square root is quite small, so expand it in a Taylor's series to three terms:

$$x = \frac{\left[-b + b \times \frac{(1 - 2 \times a \times c)}{b^2} - 2 \times a^2 \times \frac{c^2}{b^4} \right]}{(2 \times a)} \qquad (B-32)$$

With some minor modifications this becomes:

$$x = \frac{-c}{b} + a \times \frac{c^2}{b^3} \qquad (B-33)$$

With a = 0 the first term is the solution, and the second term is a first order adjustment to it. Using:

$$c = M_0 - M$$

$$b = n$$

$$a = n' \qquad (B-34)$$

the time can is then obtained from these equations through iteration.

The true anomaly currently known is for the pixel. Accuracy requires an estimate for the true anomaly of the satellite. With an estimated pixel number, the offset between the satellite and pixel true anomaly can be calculated quite precisely.

Examine Figure 24, the plane triangle of a satellite's view. The scan angle (s) is the arc between the sub-satellite point and the pixel. Only the satellite attitude angles keep the orbit plane from being normal to the triangle. The arc normal to the orbit plane from the pixel to the orbit track completes a spherical surface triangle. In orbit plane coordinates, the arc length of the normal arc (b) is the orbit plane latitude of the pixel which is known. The offset between the pixel and satellite true anomalies is the arc dv. If the attitude angles are zero, the offset is zero and the satellite true anomaly is equal to the pixel true anomaly.

To solve for the offset, one more quantity needs to be known about the spherical surface triangle. The angle γ is easy to find, as it is related to the longitude of the pixel in the satellite coordinate system.

$$\gamma = \frac{\pi}{2} - atan\,\frac{xsat}{ysat} \qquad (B-35)$$

The x and y satellite coordinates can be derived by taking the first few steps of the forward navigation algorithm (Section D), converting the pixel number to satellite coordinates. With γ known, the rules for a right spherical surface triangle give:

$$dv = asin\left[\frac{(\tan b)}{(\tan \gamma)}\right] \qquad (B-36)$$

However, the orbit plane latitude (b) and γ are both signed values, so there are four cases to be considered:

Table 4. SIGN OF DV DEPENDING ON LATITUDE AND GAMMA

Latitude	γ	sign of dv
pos	< pi/2	pos
pos	> pi/2	neg
neg	< pi/2	neg
neg	> pi/2	pos

Thus the satellite's true anomaly is:

$$vsat = vpix + dv \qquad (B-37)$$

With this final equation, the reverse navigation is well-defined.

D. FORWARD NAVIGATION

The forward navigation of an image converts the image line and pixel number of the observed pixel to its corresponding latitude and longitude. The process goes through the coordinate transformations from the satellite image coordinate system to the geodetic coordinate system, as shown previously.

Just as with the reverse navigation, there is missing information that needs to be known before the navigation can be accomplished. The position of the pixel on the Earth's surface is unknown.

The key to obtaining this unknown information is in the triangle formed by the center of the Earth, the satellite and the pixel. The satellite vector (S) from the Earth's center to the satellite is known. The direction of the view vector (V) from the satellite to the pixel is known. The pixel vector (P) from the Earth's center to the pixel is unknown. The length of the view vector can be calculated if the Earth's radius at the pixel is known. The nadir angle (n) between the view and satellite vectors is the arccosine of the dot product of those two known vectors:

$$n = acos\left[\frac{(xview \times xsat + yview \times ysat + zview \times zsat)}{r}\right] \qquad (B-38)$$

where r is the length of S (i.e., the satellite's geocentric radius). From the law of sines:

$$\frac{a}{(\sin n)} = \frac{r}{(\sin z)} \qquad (B-39)$$

where a is the Earth's radius and z is the zenith angle plus ninety degrees, and

$$\frac{v}{(\sin s)} = \frac{r}{(\sin z)} \qquad (B-40)$$

where v is the length of the view vector and s is the scan angle. The sum of s, n and v is equal to π. Combining this with equations B-39 and B-40 gives the length of the view vector V. With the view vector known, the pixel vector is given by:

$$P = S + V \qquad (B-41)$$

The Earth radius at the pixel is still unknown. Since the Earth's radius is constant within 25 km, its effect is second order and an iteration can solve the problem. This is accomplished by assuming any initial latitude for an approximation of the Earth's radius. Then iterate over the full conversion, using that latitude. When the pixel position is calculated, update the estimated latitude and iterate again, continuing until the needed accuracy is obtained. Then the forward navigation process can be completed.

APPENDIX C. CODE FOR READING WVS

```
      Program WVS
C*
C* Purpose:
c*       This purpose of this routine is to let the user select a section
c*       of WVS coastline (by latitude range).  The position of the coastline
C*       is then written to a file (lat_lon.dat) for later use or plotting.
C*       The density of the coastline (i.e., every point or every other point,
C*       etc.) can be chosen is the subroutine show_seg using the skip/ /
C*       command.
c*          Presently, this routine is in developmental form and changes
c*       are imminent.  This program worked well for the west coast of the
c*       United States.  There were problems outside of the U.S. boundaries
C*       possibly due to internaional boundaries.  This problem will be
C*       dealt with at a later time, as this program is in its early
C*       development and for our purposes, only the U.S. west coast was
C*       necessary.
C*
C* Variable naming philosphy:
C*       For the most parts the same name is given to variables in this
C*       computer code as used in WVS specifications.
C*
c* Major variable list:
c*       clat ........... Floating point equivalent of latcell.
c*       clon ........... Floating point equivalent of loncell.
c*       lrec ........... Logical record # (48 bytes).
c*       ipos ........... Used to store location of byte pointer and
c*                        pass this value to other subroutines.
c*       nbytlft ......... Number of bytes left in present physical record
c*                        to where the next cell (C) starts.
c*       nrecleft ........ Number of logical records until next cell (C).
c*       npiectoskip ..... Number of physical records to skip before getting
c*                        to next cell.
c*       prec ........... Physical record # (9600 bytes).
c*
c* Problems:
c*       The routine as been tested on any general cases. Only one data tape
c*       "as utilized. The data tested didn't have any extra attributes, etc.
c*         In addition, the bad programming  technique of common blocks was
c*       utilized for my convenience.
c*
      implicit none
      integer*4 blksize
      parameter (blksize = 9600)
      integer*4 cellnum, lrec, i,j,channel,   istat,sizeread
      integer*4 datadec,    ibeg,         latcell,   lngcell, ipos
      integer*4  clat,clng, cellstart
      integer*4 minlat,maxlat
      integer*4 nfea rec,   ncellhead,   nrec left, nseg rec
      integer*4 nleft, ;    nprectoskip, ntext,      origdec
      integer*4 nfeaincell, nsegincell
      integer*4 bytepos,    size, status
      logical   tape
      external bytepos
      character*9600 chrbuf
      character ans*1, ass*10,format*11,cellflag*1
      byte buf(9600)
      logical echo, dataflag
      equivalence (chrbuf,buf)
      integer prec
      common /physrec/chrbuf, prec
      data prec/0/, nleft/0/
      data dataflag/.true./
c
c.... define functions
      integer*4 skip_precs
c
      print '(a,$)',' Read data from tape (t) or disk (d) =>'
      read '(a)',ans
      if (ans .eq. 't' .or. ans .eq. 'T')then
         call mtopen(channel, istat)                  ! open tape device
         if (istat .eq. 1) call mtrwnd(channel, istat) ! rewind tape drive
         if (istat .ne. 1)then
            print *,' open status =',istat
            stop 'Problem opening tape drive: aborting'
         else
            print *,' Routine opened taped channel successfully'
         endif
      else
         open (unit=10,file='disk$11:[scratch.motell]vvs.dat',
     1   ACCESS='DIRECT',form='UNFORMATTED',recl=blksize/4,
     2   iostat=status,status='OLD')
         channel = 10
         if (status .ne. 0)print *,' Status problem opening diskfile'
      endif
c
      print '(a)',' User Has option to see every bit of data. But'
      Print '(a)',' I would advise you to just say NO to check data'
      print '(a,$)',' Check data (y/n) =>'
      read '(a)',ans
      if (ans .eq. 'Y' .or. ans .eq. 'y')then
        print *,' enter cell number start:'
        echo = .true.
        read *,cellstart
      else
        echo = .false.
      endif
      print *,' Echo value:'
      print *,echo
c
         print '(a,/,a,/,a,$)',
     1   ' Enter latitudes range(positive for North)',
     2   ' For example, =>32, 38',
     3   ' Enter latitudes  =>'
         read *,minlat,maxlat
         print *,' Using following lat,lon bounds:',minlat,maxlat
c
      open (unit=12,file='lat_lon',access='sequential',
     &       form='formatted',status='new')
      open (unit=15,file='SHORE.DAT',access='sequential',
     &       form='formatted',status='new')
      print '(a)',' Opening file <lat_lon.DAT> for data output'
c
c.... read at least the first four file header logical records
c
      print *,' reading first physical record'
      call get_data(channel, echo)
c
c.... decode first file header record
c
      print 700,chrbuf(1:48)
      read (chrbuf(1:48),1000)origdec,datadec
 1000 format(38x,i5,i5)
      print *,' origdec =',origdec
      print *,' datadec =',datadec
c
c.... decode second file header
c
      print 700,chrbuf(49:97)
c
c.... decode third file header
c
      print 700,chrbuf(98:145)
c
c.... decode fourth file header
c
      print 700,chrbuf(146:193)
      read (chrbuf(146:193),1001)ntext
 1001 format(41x,i4)
      print *,' ntext =',ntext
      print 705,ntext
  705 format(' Skipping <',i5,'> text records')
c
      i = 193
C
C.... Read Another physical record from tape:
C
    5 continue
      if (prec .gt. 1)then
        print 830,prec+1
  830   format(' reading physical record #',i5)
        call get_data(channel, echo)
      endif
c
c.... process data
c
   10 continue            ! increment pointer to data within physical record
c
c.... Perform data search: first check to see if lat & lon are in bounds,
c.... if not jump to the next cell. Use information in cell header to
c.... determine how many logical records must be skipped before next cell.
c
  715    format(1x,a)
         read(chrbuf(i:i),1025)cellflag
         do while (cellflag .ne. 'C')
           print 716,chrbuf(i:i+47)
  716      format(' cell mismatch:',a)
           i = bytepos(channel,i,1,echo)
           read(chrbuf(i:i),1025)cellflag
         end do
 1025    format(a1)
c
         read(chrbuf(i:i+47),1010)cellflag,ncellhead,cellnum,
     &             lngcell,latcell,
     &             nfeaincell,nsegincell,nfea rec,nseg rec
 1010    format(a1,1x,i1,i6,i8,i7,i5,i5,i7,i7)
  706    format(6x,a)
         if (cellflag .eq. 'C' .and. nfearec .ne. 0)then
            clng = lngcell/origdec
            clat = latcell/origdec
            if (clat .ge. minlat  .and. clng .le. maxlat)then
```

```
               print 715,chrbuf(i:i+47)
               call shov_fea(cellnum, channel,i, clat, clng, datadec,
     &                        echo, nfeaincell, nsegincell)
               print 715,chrbuf(i:i+47)
               if (i .gt. 9600)then
                 i = 1
                 goto 5                ! read nev physical record
              endif
              goto 10                  ! process next logical record
            else
c
c.... Determine # logical records to skip to next
c.... Must alvays move byte pointer "i" at least one logical record.
c
              nrec left = ncellhead+nfea rec+nseg rec
c
c.... Skip unvanted logical records by incrementing pointer (i)
c
              i = byte pos(channel, i, nrec left+1,echo)
              IF(I.EQ.0)THEN
                PRINT *,' I = 0  HERE'
                GOTO 5
              endif
              goto 10
            endif
         endif
         i = bytepos(channel,i,1,echo)       ! point to next logical record
         goto 10                        ! process next logical record
c
  700 format(1x,a)
      end
c
c
c
      subroutine get_data(channel, echo)
c*
c* Purpose:
c*      Routine reads one physical record from tape or disk. Assumes
c*      that logical device unit or channel = 10, vhen its from disk.
c*      If an installation every has a tape unit device number of "10",
c*      this routine vould become confused.
c*
c* Argument list:
c*      istat ......... Status of tape read (1=success).
c*
c* Problems: Uses common block vhich is a bad programming practice.
c*
      integer*4 blksize
      parameter (blksize = 9600)
      integer*4 channel, istat, sizeread
      character*9600 chrbuf
      byte buf(9600)
      equivalence (chrbuf,buf)
      integer prec
      logical echo
      common /physrec/chrbuf, prec
c
      prec = prec+1
      if (mod(prec,10).eq.0)print *,' Physical record #',prec
      if (channel .ne. 10) then
```

```
        print *,' Reading from tape '
        call mtread(channel,buf,blksize,istat,sizeread)
        if (sizeread .ne. blksize)then
          print *,' Number of bytes read =',sizeread
          print *,' Should have read     =',blksize
          print *,' ? what''s up ?'
          prec = prec-1
        endif
      else
        read (channel,REC=prec,ERR=99)buf
      endif
c
      if (echo)then
      do i=1,9600,48
        print 716,i,chrbuf(i:i+47)
        write (15,716)i,chrbuf(i:i+47)
      end do
      endif
  716 format(' G.D. ',i4,1x,a)
      return
   99 print *,' error reading record:',prec,channel
      return
      end
c
c
c
      subroutine show_fea(cellnum, channel,i, clat, clng, datadec,
     &                     echo, nfeaincell, nsegincell)
c*
c* Purpose: This subroutine is used to decode information from the feature
c*          record. After needed information is decoded it is written to a file,
c*          via other subroutines.
c*
c* Major Variables:
c*      contl_arr ....... This array stores the edge code (see pp. 14-15 of
c*                        WVS) as a function of "feanum".
c*
c*      feaflag ......... Character variable that should be 'FEA' when
c*                        decoded.
c*      feanum .......... Don't understand what this is yet.
c*      nfdatarec ....... The # of feature data records that follow the
c*                        feature header, usually 1.
c*      segdir_arr ...... This array stores the direction of the segment
c*                        (either "F","R" or "D"). The direction is stored
c*                        as a function of segment ID.
c*      seg_assoc ....... This array stores the feature number "associated"
c*                        with a given segment. Thus, the indice of "seg_assoc"
c*                        is the segment number, and the value is the
c*                        associated "feature number".
c*
c* Algorithm:
c*      1) The cell header gives the total number of feature records. The
c*         first step, therefore, is to process all feature records.
c*      2) When processing feature records, store in array the "edge code"
c*         associated with a given feature number.
c*      2) Next, the number of segment records is used as a loop limit. That
c8         is, process each group of segment records (and associated vertices)
c*         until no segment records are left. Note that there can be more
c*         segment records than feature records.
c*
```

```
c* Notes: Manual overide. It has been found that occasional the data has
c*        the incorrect number for nfdatarec or Number of Feature DATA RECords.
c*        Thus, the routine skips to where it thinks the next feature or
c*        segment record should reside and finds a different record than
c*        expected. This routine allovs one to manual overide this problem.
c*
      implicit none
      character*9600 chrbuf
      integer prec
      common /physrec/chrbuf, prec
      integer*4 cellnum, clat,clng, channel, datadec
      integer*4 i
      logical echo
c
c.... local variables
      integer segnum_arr(300), index, kk, seg_int(6)
      integer cont1, cont2, cont1_arr(300), cont2_arr(300)
      character*1 segdir_arr(300), seg_chr(6)
      character feaflag*3, featype*1,facs*5,segdir*1
      integer*4 fea_left, bytepos,nfeaincell,feanum,nseginfea
      integer*4 nattr, nfdatarec
      integer*4 nsegincell, nsdatarec, loop, segnum
      integer*4 k,kstart,j


c
c.... increment byte pointer to next record vhich should be a feature record:
c
      if (echo)then
        print *,' i,nfeaincell,nsegincell,prec',i,nfeaincell,
     &             nsegincell,prec
        print 715,chrbuf(i:i+47)
      end if
c
      fea_left = nfeaincell
      i = bytepos(channel,i,1,echo)
      kstart = 1
      do while (fea_left .gt. 0)
        fea_left = fea_left - 1
        if (echo) print 715,chrbuf(i:i+47)
        read(chrbuf(i:i+47),1000)feaflag,feanum,featype,
     &                     cont1,cont2,nseginfea,nattr,nfdatarec
c
c... Store values need for later in arrays:
c
        cont1_arr(feanum) = cont1
        cont2_arr(feanum) = cont2
 1000   format(a3,i7,a1,5x,18x,i1,i1,4x,i3,i3,i2)
        if (echo)then
          print 800,feaflag,feanum,featype,cont1,cont2,
     &              nseginfea,nattr,nfdatarec
  800     format(' feaflag=',a3,' feanum=',i2,' featype=',a1,
     &           ' cont1=',i1,' cont2=',i1,' nseginfea=',i4,
     &           ' nattr =',i2,' nfdatarec=',i3)
        endif
c
        if (feaflag .ne. 'FEA')then
          print *,' Next record is not a feature record'
          print 715,chrbuf(i:i+47)
          stop ' Program aborting !!!'
        endif
```

```
c
c.... At this point the previous record must of been a feature record
c.... Now skip past extra attribute records:
c
        i = bytepos(channel,i,nattr+1,echo)   ! skip feature data record.
c
c.... Decode feature data records and store direction and number in arrays.
c

        do j=1,nfdatarec                       ! loop for each logical record
          read(chrbuf(i:i+47),1050)(seg_int(k),k=1,6)
 1050     format(6(i7,1x))
          read(chrbuf(i:i+47),1060)(seg_chr(k),k=1,6)
 1060     format(6(7x,a1))
c
c.... store segment identification number only for actual numbers:
c
          do k=1,6
            if(seg_int(k) .ne. 0)then
              kstart = kstart+1
              index = seg_int(k)
              segdir_arr(index) = seg_chr(k)
            endif
          enddo
c
          i = bytepos(channel,i,1,echo)
          if (kstart .ge. 194)stop ' kstart is too large '
        end do
c
c.... point to first segment record:
c
    5   continue
        if (echo) print 715,chrbuf(i:i+47)                         ! ECHO
  715   format(1x,a)
      end do
c
C.... The following records are SEGment records:
C.... Pointer should be positioned at segment header record.
c
      do loop=1,nsegincell
        if(echo)print *,' loop=,nsegincell',loop,nsegincell
        call show_seg(cellnum, channel, clat,clng,cont1_arr,cont2_arr,
     &                 datadec,echo, i,loop,segdir_arr)
      end do
      return
      end
c*
c*
c*
      subroutine show_seg(cellnum, chn,clat,clng,cont1_arr,cont2_arr,
     &                       datadec,echo,i,segid,segdir_arr)
c*
c*
c* Algorithm:
c*      1) Set value of seg_verts = 0, this value will then be
c*         incremented until termination of routine or until seg_verts
c*         is greater than some preset number of the lat,lon data arrays
c*         (i.e., usually 5000).
c*      3) Start a loop to go through every segment record.
c*      2) Use parameter "skip" to determine the number seg_verts
```

```
c*        that will be written to disk. This parameter is used since the
c*        resolution of the world vector shoreline is much greater than
c*        needed. By modifying "skip", the user can skip vertices, thus
c*        decreasing the WVS resolution.
c*     x) "move_ptr" is used to move pointer past the blank segment vertices
c*        which appear at the end of a group of segment records.
c*
c* Problems: Routine only works properly when skip=4, this is because
c*           nrecs_left parameter is not decremented properly otherwise.
c*
      implicit none
      character*9600 chrbuf
      integer cellnum, prec
      common /physrec/chrbuf, p ec
      integer*4 clat,clng,cont1_arr(*),cont2_arr(*),segid
      character*1 segdir_arr(*)
      integer bytepos
      integer chn,datadec,ptr, segnum, nverts, nsdatarec, nxfearecs
c
c.... Define local variables:
c
      character segflag*3
      integer*4 feanum, iflag
      integer*4 i,j,n,nrecs_left, nverts_left, seg_verts,ians
      integer*4 counts, istat, move_ptr, skip, xseconds,yseconds
      real*4 invdiv, latarr(30000), lonarr(30000)
      logical echo, beg_seg
      data seg_verts/0/, skip/2/
      save seg_verts
c
c
      beg_seg = .true.
      counts = 0
      if (echo) print 715,chrbuf(i:i+47)
  715 format(1x,a)
      read(chrbuf(i:i+47),1030)segflag,segnum,nverts,nxfearecs,
     &                         nsdatarec,feanum
 1030 format(a3,i7,i5,2x,i2,i5,i7)
c
c.... According to previous assumptions, segid = segnum
      if (segid .ne. segnum)then
        print *,' segid ',segid,' does not equal segnum ',segnum
        return
      endif
C
C.... IF THERE ARE EXTRA SEGMENT RECORDS SKIP THEM:
C
      i = bytepos(chn, i, 1+nxfearecs,echo)
c
c.... Pointer (ptr) should be positioned at first vertice pair:
c
      nrecs_left = nsdatarec
      invdiv = 1./(datadec*3600.)
c
      if (echo) print 715,chrbuf(i:i+47)
      do while(nrecs_left .gt. 0)
c
c.... Calculate number of vertice records left:
c
        nrecs_left = nrecs_left - 1
```

```
         nverts_left = nverts-counts
c
c.... Determine how many vertices to process:
C
         n  = 4                                           ! maximum vertices per record
         if (nverts_left .lt. 4)then
           n = nverts_left
           print *,' nverts_left=, counts',nverts_left,counts
         endif
         move_ptr = (4 - n)*12
c
         do j=1,n                               ! process four or less vertices/record
           counts = counts + 1
           if (mod(counts,skip) .eq. 0)then
             seg_verts = seg_verts+1
c             if (seg_verts .gt. 10000)stop ' seg_verts = 10000'
c             if (seg_verts .gt. 7500)stop ' normal termination '
c               call closeup ( counts, seg_verts,
c     &                 latarr(1), lonarr(1), latarr(5000),
c     &                 lonarr(5000)
c             endif
             print 715,chrbuf(i:i+11)
             read (chrbuf(i:i+11),1000)xseconds,yseconds
 1000        format(i6,i6)
             latarr(seg_verts) = clat + float(yseconds)*invdiv
             lonarr(seg_verts) = clng + float(xseconds)*invdiv
             if(beg_seg)then
                iflag = 0
             else
                iflag = 1
             endif
             print *,i,
     1            seg_verts,latarr(seg_verts),lonarr(seg_verts),
     2            iflag,
     3            cont1_arr(feanum),cont2_arr(feanum)
c
             write(12,700)
     1            cellnum, seg_verts,latarr(seg_verts),
     2            lonarr(seg_verts), iflag,
     3            cont1_arr(feanum), cont2_arr(feanum)
  700        format(2x,I6,1x,i5,f10.5,1x,f10.5,1x,3(i1,1x))
             beg_seg = .false.
           endif
           i = i+12
           if (i .ge. 9600)then
             call get_data(chn, echo)
             i = 1
           end if
         end do
         i = i + move_ptr
         if (i .eq. 9600)print *,' ========== i =',i
      end do
      return
      end
c*
c*
c*
      integer function skip_precs(channel,nprectoskip,echo)
c
      implicit none
      integer prec
      character*9600 chrbuf
      common /physrec/ chrbuf, prec
      integer*4 nprectoskip,j,istat,channel,nleft
      INTEGER I
      logical echo
c
      do 30 j=1,nprectoskip
         call get_data(channel,echo)
   30 continue
      skip_precs = prec
      return
      end
c*
c*
c*
      integer function bytepos(channel, pointer, lrec, echo)
c
c* Purpose:
c*     This routine is used to move the "byte pointer" to where
c*     the next logical record needed is located. By setting the logical
c*     record size to zeroe, this routine can be used to place pointer in
c*     in next logical record. When lrec is greater than zeroe this routine
c*     will move the pointer to either the correct location in the present
c*     physical record, or the needed location in the next physical record
c*     according to where the next needed data should be located.
c*
      implicit none
      character*9600 chrbuf
      integer prec
      logical echo
      common /physrec/ chrbuf, prec
      integer*4 channel, pointer, lrec, prectoskip, bytes_left
      integer*4 istat,skip_precs, i
      external skip_precs
c
      bytes_left = lrec*48 + pointer
      if (bytes_left .lt. 9600)then
        bytepos = bytes_left
        return
      else
c
c.... check to see if whole physical records need to be skipped
c
        bytes_left = bytes_left - 9600       ! # bytes in present record
        prectoskip = bytes_left/9600         ! # physical records to skip
        if(prectoskip .gt. 0)then
          prec =  skip_precs(channel,prectoskip,echo)
c
c.... set pointer to new value in new physical record
c
          bytes_left = bytes_left - prectoskip*9600
        end if
      endif
c
      call get_data(channel, echo)
      bytepos = bytes_left
c
      return
      end
```

```
c*
c*
c*
      subroutine closeup(prec,n,lat1,lon1,lat2,lon2)
      integer prec,n
      real lat1,lon1,lat2,lon2
c
      print *,' Program processed:',n,' vertices at prec: ',prec
      print *,' lat,lon range from:',lat1,lon1,' to ',lat2,lon2
      close (15)
      close (12)
      close (10)
      stop ' normal termination '
      end
```

REFERENCES

Anuta, P. E. and Farzin Davallou, 1982. Resolution Matching for Registration of Dissimilar Images. *Proc. of PRIP 82. IEEE Comp. Soc. Conf. on Pattern Recognition and Image Processing 14-17,* June 1982, Las Vegas, NV, 327-332.

Anuta, P. and C. McGillem, 1986. Parameter Space Techniques for Image Registration. *IGARSS '86. Remote Sensing: Today's Solutions for Tomorrow's Information Needs,* August 1986 (ESA SP-254), **2**, 989-994.

Barnea, Daniel I. and Harvey F. Silverman, 1972. A Class of Algorithms for Fast Digital Image Registration. *IEEE Transactions on Computers,* **C-21**, 179-186.

Bethke, W. J., 1988. Accuracy of Satellite Data Navigation. *M.S. Thesis, Naval Postgraduate School,* Monterey, CA, 87 pages.

Bowditch, Nathaniel, 1984 ed. American Practical Navigator: An Epitome of Navigation. Defense Mapping Agency Hydrographic/Topographic Center, Washington, D.C.

Brunel, P. and A. Marsouin, 1987. An Operational Method Using ARGOS Orbital Elements for Navigation of AVHRR Imagery. *International Journal of Remote Sensing,* **8**, 569-578.

Brush, R. John H., 1985. A Method for Real-Time Navigation of AVHRR Imagery. *IEEE Transactions on Geoscience and Remote Sensing,* **GE-23**, 876-886.

Brush, R. J. H., 1988. The Navigation of AVHRR Imagery. *International Journal of Remote Sensing,* **9**, 1491-1502.

Burks, Doug, 1988. NPS Satellite Image Navigation. (Unpublished manuscript, Naval Postgraduate School, Monterey, Ca).

Cordan, Earnest W., Jr. and Benjamin W. Patz, 1979. An Image Registration Algorithm Using Sampled Binary Correlation. *1979 Machine Processing of Remotely Sensed Data Symposium, IEEE,* 202-211.

Crombie, M. A., 1983. Coordination of Stereo Image Registration and Pixel Classification. *Photogrammetric Engineering and Remote Sensing,* **49**, 529-532.

Davis, W. A. and S. K. Kenue, 1978. Automatic Selection of Control Points for the Registration of Digital Images. *Proc. of the 4th International Joint Conference on Pattern Recognition 7-10 November 1978,* Kyoto, Japan, 936-938.

Defense Mapping Agency, 1988. Product Specifications for World Vector Shoreline. DMA Hydrographic/Topographic Center, Washington D.C.

Duda, Richard O. and Peter E. Hart, 1973. Pattern Classification and Scene Analysis. John Wiley and Sons, Inc., New York, 482 pages.

Emery, William J., Jim Brown and Z. Paul Nowak, 1989. AVHRR Image Navigation: Summary and Review. *Photogrammetric Engineering and Remote Sensing*, **55**, 1175-1183.

Emery, W. J. and M. Ikeda, 1984. A Comparison of Geometric Correction Methods for AVHRR Imagery. *Canadian Journal of Remote Sensing*, **10**, 46-56.

Escobal, Pedro Ramon, 1965. Methods of Orbit Determination. John Wiley and Sons, Inc. New York 463 pages.

Eversole, William L. and Robert E. Nasburg, 1983. Maximum Likelihood Estimation for Image Registration. *Applications of Digital Image Processing VI, Proceedings of SPIE-The International Society for Optical Engineers*, **432**, 190-194.

Fram, Jerry R. and Edward S. Deutsch, 1975. On the Quantitative Evaluation of Edge Detection Schemes and Their Comparison with Human Performance. *IEEE Transactions on Computers*, **C-24**, 616-627.

Gilmore, J. F., 1983. Image Registration through the Exploitation of Perspective Invariant Graphs. *Proc. SPIE Applications of Digital Image Processing 19-22 April 1983*, **397**, 55-63.

Goshtasby, A., S. H. Gage and J. F. Bartholic, 1984. A Two-Stage Cross Correlation Approach to Template Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-6**, 374-378.

Goshtasby, A., G. C. Stockman and C. V. Page, 1986. A Region-Based Approach to Digital Image Registration with Subpixel Accuracy. *IEEE Transactions on Geoscience and Remote Sensing*. **GE-24**, 390-399.

Gupta, J. N. and Paul A. Wintz, 1975. A Boundary Finding Algorithm and its Applications. *IEEE Transactions on Circuits and Systems*, **CAS-22**, 351-362.

Hall, Ernest L., 1979. Computer Image Processing of Remotely Sensed Data. Academic Press, Inc., New York, 256 pages.

Hamilton, Walter Clark, 1964. Statistics in Physical Science:Estimation, Hypotheses Testing and Least Squares. The Ronald Press Company, New York.

Hendersen, T. C., E. E. Triendl and R. Winter, 1985. Edge- and Shape-Based Geometric Registration. *IEEE Transactions on Geoscience and Remote Sensing*, **GE-23**, 334-342.

Ho, Diem and Adel Asem, 1984. Automatic Registration of METEOSAT and NOAA AVHRR LAC Images. *Proceedings of Remote Sensing Society Conference, 10th*, 81-90.

Ho, Diem and Adel Asem, 1986. NOAA AVHRR Image Referencing. *International Journal of Remote Sensing*, **7**, 895-904.

Hord, Michael R., 1982. Digital Image Processing of Remotely Sensed Data. Academic Press, Inc., New York, 256 pages.

Jayroe, R. R., J. F. Andrus and C. W. Campbell, 1974. Digital Image Registration Method Based Upon Binary Boundary Maps. *National Aeronautics and Space Administration*, NASA-TN-D-7607:m-121.

Kashef, Bayesteh G. and Alexander A Sawchuk, 1983. A Survey of New Techniques for Image Registration and Mapping. *Applications of Digital Image Processing VI, Proceedings of SPIE-The International Society for Optical Engineers*, **432**, 222-239.

Labovitz, M. L. and J. W. Marvin, 1986. Precision in Geodetic Correction of TM Data as a Function of the Number, Spatial Distribution and Success in Matching of Control Points: A Simulation. *Remote Sensing of Environment*, **20**, 237-252.

Leberl, Franz W. and Walter Kropatsch, 1980. Automated Registration of Scanned Satellite Imagery With a Digital Map Data Base. Tech. Report, Institute for National Surveying and Photogrammetry, Technical University, Austria, September 1980.

Legeckis, R. and J. Pritchard, 1976. Algorithm for Correcting the VHRR Imagery for Geometric Distortions due to the Earth Curvature, Earth Rotation and Spacecraft Roll Attitude Errors. NOAA Technical Memorandum, NESS 77, Washington D.C., 31 pages.

Moik, Johannes G., 1980. Digital Processing of Remotely Sensed Data. National Aeronautics and Space Administration, NASA SP, 431 pages.

Nack, M. L., 1977. Rectification and Registration of Digital Images and the Effect of Cloud Detection. *4th Annual Symposium on Machine Processing of Remotely Sensed Data 21-23 June 1977*, West Lafayette, Indiana, 12-23.

Ng, K. Y. K., 1977. An Automatic Image Registration and Overlay System. *Computer and Electrical Engineering*, **4**, 71-85.

Novak, Leslie, 1978. Correlation Algorithms for Radar Map Matching. *IEEE Transactions on Aerospace and Electronic Systems*, **AES-14**, 641-648.

Orti, F., 1981. Optimal Distribution of Control Points to Minimize Landsat Image Registration Errors. *Photogrammetric Engineering and Remote Sensing*, **47**, 101-110.

Pavlidis, Theo, 1982. Algorithms for Graphics and Image Processing. Computer Science Press, Inc., Rockville Md, 416 pages.

Pratt, William K., 1974. Correlation Techniques of Image Registration. *IEEE Transactions on Aerospace and Electronic Systems*, **AES-10**, 353-358.

Rao, P. Krishna, Susan J. Holmes, Ralph K. Anderson, Jay S. Winston and Paul E. Lehr, editors, 1990. Weather Satellites: Systems, Data, and Environmental Applications. American Meteorological Society, Boston MA, 503 pages.

Rauste, Y. and Risto Kuittinen, 1985. Digital Rectification of Satellite Images *Image Science '85*, Helsinki, Finland, 15-18.

Rosenfeld, Azriel and Gordon J. Vanderbrug, 1977. Coarse-Fine Template Matching. *IEEE Transactions on Systems, Man and Cybernetics*, 104-107.

Saufley, Duane C., 1982. Navigation of Geosynchronous Satellite Images. CIRA (Cooperative Institute for Research in the Atmosphere), Colorado State University.

Smith, Eric A., 1980. Orbital Mechanics and Analytic Modelling of Meteorological Satellite Orbits. Paper no. 321, Department of Atmospheric Science, Colorado State University, 161 pages.

St. Pierre, David B., 1988. Automa'-d Earth Location of Satellite Data: Recent Advances in Image Registration ıhrough Pattern Recognition and Matching. (Unpublished manuscript, Naval Postgraduate School, Monterey Ca).

Stockman, G., S. Kopstein and S. Benett, 1982. Matching Images to Models for Registration and Object Detection via Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4, 229-241.

Sullivan. D. R. and J. S. Martin. 1981. A Coarse Search Correlation Tracker for Image Registration. *IEEE Transactions on Aerospace and Electronic Systems*, AES-17, 29-34.

Sun, Hanfang, 1981. Image Registration by Combining Feature Matching and Gray Level Correlation. Technical Report, Maryland University, College Park, Computer Vision Lab, 8 pages.

Vanderbrug, Gordan J. and Azriel Rosenfeld, 1977. Two-Stage Template Matching. *ILEE Transactions on Computers*, C-26, 384-393.

Wong, Robert Y. and Ernest L. Hall, 1979. Performance Comparison of Scene Matching Techniques. *ICEL Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1, 325-330.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	C. H. Wash (Code MR/WX) Department of Meteorology Naval Postgraduate School Monterey, CA 93943-5000	8
4.	P. Durkee (Code MR/DE) Department of Meteorology Naval Postgraduate School Monterey, CA 93943-5000	1
5.	C. Motell (Code MR/ML) Department of Meteorology Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Chairman (Code OC/CO) Department of Oceanography Naval Postgraduate School Monterey, CA 93943-5000	1
7.	James R. Cherry (Code OC/CH) Department of Oceanography Naval Postgraduate School Monterey, CA 93943-5000	1
8.	N. T. Garfield (Code OC/GF) Department of Oceanography Naval Postgraduate School Monterey, CA 93943-5000	1
9.	J. Nystuen (Code OC/NY) Department of Oceanography Naval Postgraduate School Monterey, CA 93943-5000	1
10.	Brian C. Spaulding 702 N. Argonne Ave. Sterling, VA 22170	3

11. Commanding Officer 1
Naval Ocean Research and Development Activity
Stennis Space Center
MS 39529-5004

12. Chief, Hydrographic Programs Division 1
Defense Mapping Agency (Code PPH)
Bldg. 56, U.S. Naval Observatory
Washington, DC 20305-3000

13. Director (Code HO) 1
Defense Mapping Agency Hydrographic/Topographic Center
6500 Brookes Lane
Washington, DC 20315-0030

14. Morris Glenn MCPE 1
Defense Mapping Agency Hydrographic/Topographic Center
6500 Brookes Lane
Washington, DC 20315-0030

15. Director, Charting and Geodetic 1
Services (N/CG)
National Ocean and Atmospheric Administration
Rockville, MD 20852

16. Ann Miers SC/SGE 1
Defense Mapping Agency Hydrographic/Topographic Center
6500 Brookes Lane
Washington, DC 20315-0030

17. Dr. Chris Crosiar 1
Navy Oceanographic and Atmospheric Research Lab
Monterey, CA 93943-5005

18. Commanding Officer 1
Fleet Numerical Oceanographic Center
Monterey, CA 93943-5005

19. Dr. John Hovermale, Director 1
Atmospheric Directorate
Navy Oceanographic and Atmospheric Research Lab
Monterey, CA 93943-5005